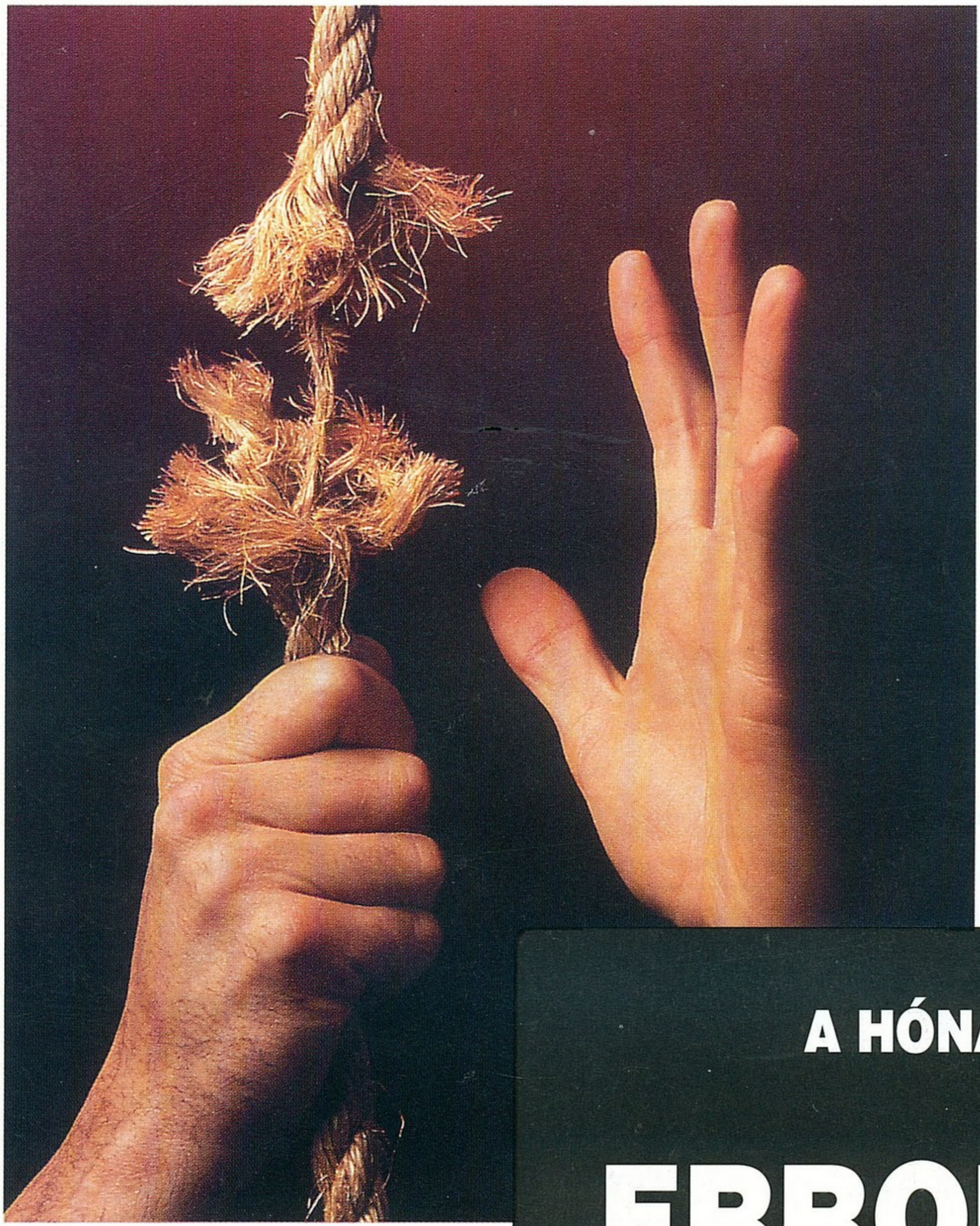
XV. ÉVF. 5. SZÁM, 1997. MÁJUS

EXTRA CD-VEL ÁRA: 586 FT

ÚJ ALAPLAP

150. SZÁM

SZÁMÍTÁSTECHNIKAI FOLYÓIRAT LEMEZMELLÉKLETTEL



A CD-ROM-on:

- Merlin - interaktív demó
- Multimédia nyelvoktatás
- Office 97 dokumentumok
- Platinum szoftverek
- Epson driverek
- MQSeries

- Emulált MSCDEX
- Helllp! - a helpkészítő
- Assembly ujjgyakorlatok
- DLL-adminisztrátor
- Topográfia vizsga

A HÓNAP TÉMÁJA:

ERRORIZMUS

A kukac

Nélkülözhetetlen fájlkezelők

Öt operációs rendszer egy gépen

ÚJ ALAPLAP

A Mikroszámítógép Magazin és az Alaplap hagyományait folytató számítástechnikai folyóirat

Megjelenik havonta, mágneslemez melléklettel

Főszerkesztő:
Faklen Pál

Főszerkesztő-helyettes:
Varga János

Szerkesztő:
Jakab Ágnes

A szerkesztőbizottság tagjai:
Aszalós László, Csórián Sándor, Feleki Zoltán, Ferenczi Gábor, Herczeg József, Horlai János, Kis János, Nagy Gábor, Pogány Csaba, Szondi Egon János, Vargha Dénes, Vékony Tamás

Szerkesztőség és kiadó:
1539 Budapest I., Márvány u. 17.
Telefon: 156-3211 / 200, 201
Fax: 214-9492
E-mail: alaplap@mail.datanet.hu

Felelős kiadó:
Faklen Pál

Terjesztés:
Megyes Zsuzsanna

Hirdetésszervezés:
Árvai Katalin, Bogácsi Mária, Galyasi Hedvig, Pap Katalin

Külföldi hirdetések:
PubliCity
Reklám- és Médiaügynökség
1537 Budapest I., Márvány u. 17.
Telefon: 156-1182 Fax: 175-3539

A kiadó a hirdetések tartalmáért és a nyomdakészen kapott hirdetések formájáért (és helyesírásáért) nem vállal felelősséget

Példányszámadatok hitelesítése:
Magyar Terjesztésellenőrző Szövetség
MATESZ
Ez a szám
11 000 példányban jelent meg

Nyomtatás:
Zalai Nyomda Rt, Zalaegerszeg
Felelős vezető:
Somogyi Tibor ügyvezető igazgató

Terjeszti:
A Magyar Posta Rt, a Nemzeti Hírlapkereskedelmi Rt, a Hírker Rt, a Kiadói Lapterjesztő Kft, számos számítástechnikai szaküzlet és más alternatív terjesztő

Előfizethető a kiadónál:
Új Alaplap Kiadói Kft,
1539 Budapest, Pf. 571
Bankszámlaszám:
OTP 11701004-20171649
Eladási ár: 396 Ft (E szám 586 Ft)
Évi előfizetési díj: 4356 Ft

Évi külföldi előfizetés díja:
4356 Ft + postázási költség

HU ISSN 1217-7598

A HÓNAP TÉMÁJA: ERRORIZMUS

(Összeállította: Vargha Dénes)



Az errorizmus lélektana

A hardverekben fellépő hibákról ezúttal nem beszélünk. Azokat a szervizek (általában) el tudják hárítani, ha másképp nem, hát az alkatrészek szisztematikus cseréjével. Csak tapasztalt szervizesek és garantáltan működő pótalkatrészek kellenek hozzá. De mi van a szoftverekben? Hiba. Azt hiszem, hiba nélkül nem is képzelhető el a számítástechnika. Hogy ez hiba? Lehet, hogy az, mert ugye a kibernetika egzakt tudomány. Mégis arra kell berendezkednünk, hogy hosszú távon együtt tudjunk élni a hibákkal. Tudjuk kezelni azokat. Ha egyszer már vannak. De miért is vannak?

A hibák magyarázata nagyon prózai: a számítástechnika emberi. Emberi annyiban, hogy emberek hozzák létre, akik tervezéskor (tételezzük fel) minden eshetőségre kiterjedő, totális rendszertervet alkottak, de ugye a megvalósítás során óhatatlanul becsúszó hibákat már nem tudják teljesen kiszűrni. Hogy miért? Nos, egy termék csupán félkésznek tekinthető mindaddig, amíg át nem esett a végeláthatatlan tesztek, nyúzások procedúráján. Ez a tesztfázis azonban rendkívül nagy odafigyelést, sok időt és embert, vagyis rengeteg pénzt követelne meg. Nem csoda, ha sokan megspórolják. Inkább nyilvános bétatesztelési akciókkal próbálják áthárítani a felelősséget a széles felhasználótáborra.

Bocsánatos bűn-e az, ha egy erőltetett menetben kiadott n+1-edik verzió többször tíz új kényelmi funkcióval kápráztat el ugyan mindenkit, de az elkapkodott megjelentetés miatt tartalmaz néhány kolosszális és több száz banális hibát is. Sokan úgy gondolják, sebaj, majd jön egy javítás, update vagy patch, és szép lassanként kigyomlálható minden nem odavaló vagy elrontott programkód.

Megkövetelhető-e a hibátlanság?

Igen, de követeljük is meg. Hogyan? Bojkottáljuk, ne használjuk, ne vegyük meg azokat a programokat, amelyekkel közismerten problémák vannak. Honnan szerezhetünk tudomást ezekről? Az Internetről, ismerőseinktől, a SZAKSAJTÓBÓL. Igen, kedves (szak)újságíró társadalom, itt értékelődik fel a szerepünk, ha hajlandók is vagyunk vele élni. Nem egy adott termék gyártójának, forgalmazójának, hazai képviselőjének ülepét kellene illetni ízlelőszervünkkel, — merthogy a hirdetések tartják el a lapot —, hanem bátran meg kell írni a színtiszta igazságot. Ez akár még a példányszám növekedéséhez is vezethetne. Egy őszinte, szigorúan kritikus és önkritikus laphoz persze kellően tapasztalt, hiteles

újságírók is kellenek, akik szavára hallgatnak. Hiába fordulnánk a Minőségellenőrzési Intézethez (MEI), mert elterjedt, egyértelmű követelményrendszer hiányában hiú ábránd a szoftverek objektív tesztelése.

A hardvertesztek mintájára létre kell hozni szoftvertesztkörnyezeteket is, lehetőleg „vasfüggetlen" módon. (Ez csupán annyit jelent, hogy minden hasonló célra készített szoftvert azonos hardveres feltételek mellett vizsgáljunk, lehetőleg ugyanazokon a konfigurációkon.) Ki kell dolgozni az alapvető szövegszerkesztési, táblázatkezelési, képfeldolgozási stb... műveletsorokat, lehetőleg többlépcsős változatban, vagyis legyen alapfokú, középfokú és profi követelményszint is.

A szoftverfejlesztő eszközökre pedig megfelelő alapfeladatokat kell kitűzni, majd a generált kódot górcső alá véve megnézni, hogy az mennyire rugalmas, gyors, pontos, kisméretű stb. A megtalált hibákra pedig kegyetlenül le kell csapni, azokat ki kell pellengérezni, hogy mindenki lássa: ezt és így nem szabad csinálni.

Meggyőződésem, hogy ha a fentiek pozitív fogadtatásra lelnek a felhasználók körében, és érdemben is befolyásolni fogják az eladásokat, a gyártók, illetve forgalmazók rendkívül gyorsan fognak rá reagálni. Kezdetben támadnának, majd védekeznének, később a még el sem készült termékeket kívánnák a független tesztelőgárdával véleményeztetni, végül tényleg ráállnának a korrekt, durva hibák nélküli termékek megjelentetésére. Egye fene, legfeljebb csak kétévenként lenne új Office vagy CorelDraw. Kibírnánk.

Ezek azonban egyelőre utópisztikus elképzelések. Induljunk ki a keserű valóságból. Hogyan küzdhetünk a mégiscsak meglévő, makacs hibák ellen? Lássuk például, mi is jöhet szóba egy olyan felhasználó számára, akinek munkája, pénze, egzisztenciája függ számítógéppel előállított, feldolgozott, tárolt adatainak biztonságától. Mert szerencsére nem csak olyanok „dolgoznak" számítógéppel, akik kikapcsolódásképpen szenvedélyesen merednek a villogó monitorra, és akik számára az elvesző adat legfeljebb a „high score" megsemmisülését jelentheti az „Üsd, vágd, nem apád!" nevű akciójátékban.

Magad uram...

Mivel a programhibák miatt semmivé lett vagy megsérült értékes adatállományokat senki nem pótolja, más megoldásokat kell keresni.

1) Adatmentés

Kötelező napirendi pontként szakszerűen végrehajtva. Persze nem árt néha napirenden kívül is! Sajnos a komoly, jó nevű adatmentő és archiváló készülékek, meghajtók ára három-négyszerese a harddiszkekének. Hasonlóan magas a vállalati adatarchiváló rendszerek ára a kommersz „adatszaporító" alkalmazásokéhoz viszonyítva.

2) Cölöpök leverése a tóban

Erre az ősrégi módszerre főleg a programozók kényszerülnek rá. Ahogy a rómaiak mondták, hajózni pedig kell... és ma már ugyanúgy programozni is. Hiába tartogat alattomos zátonyokat és jéghegyeket a compiler és linker, a jó és leleményes programozó kiismeri fejlesztőrendszere gyengéit, és kikerüli azokat. Baj csak akkor van, ha az adott fejlesztőeszközből újabb változat lát napvilágot, és a programíró kezdheti elölről az aknakeresést. Van, aki erre már rest. Ezért is dolgoznak olyan sokan még mindig a Clipper 87 Summer fordítóval.

3) Valódi hotline

Telefonos, később esetleg alternatív csatornaként internetes „forródrót", amelyen keresztül az elakadások okára fény derül. A bejelentett valódi hibák körültekintő (online kapcsolat esetén téves diagnózistól mentes) behatárolása és kivizsgálása, melynek nyomán a gyártó záros határidőn belül köteles elhárítani a hibát. Ehhez persze a forródrót szolgáltatói végén igazán profi, tapasztalt szakemberek kellenek. Szerintem — ha megfizetnék őket — erre is lenne alkalmas munkaerő. A hiba elhárítása akár egyszerű telefonvonalon, modemes távirányítással is megtörténhetne, ahol a bajba jutott fél sokat tanulhatna a szakember útmutatásaiból és magyarázataiból. Csak attól tartok, hogy ez a rendszer is nagyon hamar elgépiesedne. Típushibák, típusmegoldások. Mindössze az eddig még nem regisztrált hibajelenségek igényelnék az élő, emberi beavatkozást. Ki fogja megcsinálni az első ilyen független szoftverkonzultáns és hibaelhárító céget Magyarországon? Az ötlet szabad, és a nyersanyag (error) is korlátlan mennyiségben rendelkezésre áll...

4) Jogi úton

Jogi út kizárva. Gyakorlatilag ez az olvasata a szoftverek licencszerződéseinek. Egyelőre. Pedig ennek nem kellene feltétlenül így lennie. A kiszolgáltatott felhasználóknak fel kellene lépniük ellene, mindössze annyit elérve, hogy a szoftvert is kezeljék hasonlóan, mint a hardvereszközöket. Azokra bizonyos ideig legalább van garancia, és ha a termék hibás, akkor kicserélik vagy megjavítják, bizonyos esetekben pedig még a javíthatatlan termék ára is visszajár.

„Van másik!"

Akár hiszik, akár nem, a hibás szoftverek ellen is alkalmazhatók az úgynevezett hibatűrő rendszerek, bár ezek már mindenképpen fejlettebb, többprocesszoros hardverkörnyezetet feltételeznek. Az igazi fegyver a redundancia (az adattárolás és a futtatás többszörözése) és az adatok konzisztenciájának folyamatos vizsgálatával a véletlen hibák kiszűrésére. Kissé eltúlzott példával illusztrálva: ha rettegünk attól, hogy levélírás közben elszáll a „Microsoft Kiemelt Fogalmazója", a WinWord, üzemeltessünk egyidejűleg 3 gépet. Mindegyiken fusson a Word. A gépek billentyűzete és egere legyen közös, így elég lenne egyetlen klaviatúrát csépelni, és egyetlen egeret tologatni ahhoz, hogy mindhárom gépen egyformán íródjon az a bizonyos levél. Ha egyik gépünk el találna szállni, nem baj, van másik, ahogy a klasszikus zenebohóc mondotta vala...

Addig is, amíg a hibatűrő rendszerek nem terjednek el a kommersz, populáris számítástechnikában, be kell érnünk hibatűrő felhasználókkal. Ezek lennénk mi! Mi?

Nem árt persze azt is megemlíteni, hogy a látszólag megmagyarázhatatlan hibák hirtelen felbukkanásának igen gyakori oka a vírusfertőzés. Ami azonban teljes biztonsággal megelőzhető. Jobb félni, mint megijedni. Kinek drágább (adatai) rongy élete... Nos, az ne csak imádkozzon, hanem archiváljon buzgón és rendszeresen, valamint ne engedjen gépéhez és hálózatához ellenőrizetlen adathordozót. Manapság már rafinált vírusok csücsülnek Excel és Word állományokban is, nem beszélve az internetes éra legmenőbb Java kártevőiről. A vírusírás és vírusirtás újra reneszánszát éli, akárcsak 4-5 évvel ezelőtt. De ez már egy másik kimerítő cikk témája.

Vajon hiba volt-e ennek a cikknek a megírása? Sajtóhiba?!

Herczeg József

Az öngyilkosság nem a legjobb megoldás

Élet a halál után

A számítógépet használók többsége bizonyára nemegyszer átélt már olyan helyzetet, amikor valamelyik program, illetve maga az operációs rendszer „lefagyott". Ilyenkor se kép, se hang... Semmiféle életjel a billentyűzetet vagy az egeret mozgatva. A válogatott szidalmak és Bill Gates sűrű emlegetése közben nem mindig gondoljuk végig — talán mert nem is tudjuk —, hogy mi zajlott le a háttérben.

A gép lefagyását több esemény szerencsétlen összejátszása okozhatja. Az egyik a hardvereredetű problémák sorozata. Ilyenkor a gép leállításáról a BIOS gondoskodik, hogy megelőzze a hibás műveleteket. A leállás oka egyformán lehet hibás RAM — „RAM parity error" — vagy akár más hardverhiba. Ebben az esetben a processzort a BIOS a HALT paranccsal megállítja, és nem lehet mást tenni, mint újraindítani a gépet, imádkozva, hogy sikerüljön az újraindítás.

Interruptvita, gyorshajtás

Ugyanilyen furcsaságokat lehet tapasztalni akkor is, ha az egyes interruptok (megszakítások) összevesznek egymással, azaz a megfelelő szoftverek lekezelése nélkül ugyanazt az eszközt, ugyanazt a hardverinterruptot akarják használni. (Ez azonban nem vezet minden esetben rendszerleálláshoz.) A tisztázatlan eredetű bajok hátterében is sokszor „interruptvita" húzódik meg. Klasszikus példa erre az egér és a modem konfliktusa: a modemet nem lehet rendesen használni, mert ugyanazon a megszakításon ül, mint az egér.

Jellegzetes tünet az is, amikor a gép ciklikusan újraindul, elölről kezdve a RAM-tesztet, majd mire nekilátna a rendszer betöltésének, megint a BIOS végrehajtásához tér vissza. Ilyen hibákat rosszul konfigurált kártyameghajtó programok éppúgy okozhatnak, mint olyan memóriamenedzserek, amelyek összevesznek operációs rendszerünkkel. Az ok: nem létező ugrócímre mutat valamelyik végrehajtandó parancs, vagy pedig a BIOS elejére, ami a rendszer újraindulását eredményezi.

Ha az operációs rendszer elindult, akkor a hiba (elvileg) már csak a konfigurációval vagy a programmal lehet, kivéve... Nos, persze kivételek is vannak. Például olyan hardver jellegű hibák, amelyek kizárólag ilyenkor kerülnek napvilágra. Bizonyos processzorokkal semmi gond sincs, ha magasabb órajellel repesztünk a névlegesnél. De előfordulhat — például az OS/2-nél —, hogy az operációs rendszer felismeri a processzort, és megbünteti a gyorshajtásért, a biztonságos működés érdekére hivatkozva inkább leáll a rendszer is, és nem futnak a programok. Az Intel processzorainak egy részébe egyenesen bele van építve a túlhajtás elleni védelem: maga a processzor sem hajlandó elindulni a megengedettnél nagyobb órajellel.

A multitaszk besegít

Az operációs rendszerek különbözőképpen viselkednek a mindennapi gyakorlatban. A klasszikus operációs rendszerekben semmi védelem sincs a lefagyás ellen. Ezeknél bármilyen fatális rendszerhiba (akár a BIOS-tól származik, akár máshonnan) a processzor leállásához vezet. Klasszikus példa erre a QEMM-Deskview páros híres regiszteres hibaüzenete. Használatukkor már van egy minimális védelem — amolyan multitask kezdemény —, amely megpróbálja lokalizálni a hiba hatását, amit a DOS nem tesz meg. Az esetek kisebbik hányadában a Deskview lehetővé teszi a rendszer továbbfutását azzal, hogy kiiktatja a hibás taskot. De azért ez még nem egy lefagyás ellen igazán védett rendszer. Komoly védelemre csak a valóban jól megírt multitaszk multithread operációs rendszerek képesek.

A hagyományos DOS egytaszkos operációs rendszer, ahol a processzek egymás után kerülnek végrehajtásra. Ez azzal is jár, hogy bármelyik processzben következik be fatális hiba, az lehetetlenné teszi a következő folyamat megvalósítását. A valódi multitaszkos rendszerekben más a helyzet: az operációs rendszer vezénylete alatt az egyes taszkokban úgy működnek az egyes programok, mintha önálló processzorokon és gépeken futnának. Ez azt jelenti, hogy egy ilyen virtuális gépen bekövetkező hibának egyedül arra a virtuális gépre szabad kihatnia — természetesen feltételezve egy jól megírt operációs rendszert!

Az OS/2 és a Unix volt az első két olyan operációs rendszer, amely maradéktalanul megfelelt ennek a multitaszk követelménynek. Ezeknél igen nehéz is klasszikus DOS-típusu rendszerlefagyást okozni: a rendszer többi részének működéséről egy-egy task vagy thread (szál) lefagyása után is gondoskodik az operációs rendszer és az operációs rendszer magja, a kernel folyamatosan védetten fut. A kernel legfőbb feladata tulajdonképpen egy bonyolult adminisztráció: annak megállapítása, hogy éppen milyen újabb virtuális gépek, azaz valójában mely taszkok indíthatók

GAME OVER



a gép teljesítményének és a memória lehetőségeinek adott korlátai között.

Ez a folyamat egyben azt is feltételezi, hogy az egyes taszkok egymás hibáival szemben védettek, amihez viszont az kell, hogy az adatcsere és a futás szigorúan meghatározott biztonsági szinteken történjék. A Windows 95 legfőbb hibája éppen az, hogy ezt csak felemás módon valósítja meg, ezért néhány taszk viszonylag könnyen le tudja kötni akár a rendszer összes erőforrását. Tipikus példa erre, hogy amikor másolunk, az önmagában lefoglalja rendszerünket. Az OS/2 alatt ilyesmit nem tapasztalunk.

Mondhatna is valamit

Egy másik probléma a Win95-tel, hogy sokféle, de nem kellően átgondolt foltozgatást építettek bele, ami nem tett jót a biztonságnak. Hiányzik ugyanis az a mag, amely elvégezhetné a rendszer elszigetelt virtuális gépekre való elkülönítését, és az erőforrások megosztását. Ennek részben az a forrása, hogy meg akarták őrizni a kompatibilitást a korábbi, nem kimondottan multitaszkos megoldásokra tervezett programokkal. De a szokásos Microsoft-szindrómát is felismerhetjük benne: a nagy sietségben túlságosan szétdarabolták a fejlesztést, mert fontosabb volt számukra a piac mielőbbi megszerzése, mint a biztonság.

Bizonyos mértékig a Windows 95 is képes a hibák elhárítására, amennyire ezt a virtuális gép koncepciójának átvétele és egy valódi multitask jellegű operációs rendszer biztosítja. Ha maga az alkalmazói program korrekt, akkor lefagyása esetén kellően informatív hibaüzenetet kapunk, és az adott virtuális gép kiiktatható. A többi folyamat ilyenkor gond nélkül megy tovább. Baj csak akkor van, ha a kiakadt program nem szabványos felületen keresztül kommunikál a rendszerrel. Sajnos, a Microsoftnak gyakran még saját alkalmazásai sem a szabványos felületeket használják. Ezért fordulhat elő, hogy egyetlen task kiakadása nem csupán egy virtuális gép, hanem az egész rendszer összeomlását okozza.

A Symantec programozóinak becsületére legyen mondva, hogy éppen az ilyen hibák kiküszöbölésére készítettek egy teljes értékű freeware programot — ez a Shield —, amellyel nagyon sok esetben meg lehet akadályozni a Windows 95 összeomlását, mely sok esetben abból fakad, hogy az elszálló programok kiiktatásuk után is tovább foglalják az erőforrásokat, hatást gyakorolva más taszkokra. A Symantec fejlesztői ezeket az elvarratlan szálakat próbálták kezelni, és megírták a Windows 95 olyan részeit is, amelyekről a Microsoft programozói elfeledkeztek.

Legalább az oprendszer...

A komolyabb operációs rendszerek (amilyen például a Novell) nagyrészt védettek az alkalmazói programok illetéktelen manővereivel szemben. Az operációs rendszer ugyanis meghatározza, hogy egy adott program mely szinten és milyen prioritásokkal futhat. Igaz, hogy ez némiképp lassítja a programok végrehajtását, mert az operációs rendszernek a futtatott programot állandóan felügyelnie kell, de sokkal nagyobb az az előny, hogy az operációs rendszer nem fagyhat le a külső programok hibája miatt.

A Windows NT használói jól ismerik az ún. kék halál jelenségét. Ez az az eset, amikor a rendszer már nem képes adminisztrálni és kijavítani a működése közben fellépő hibákat, ezért az adatkárok elkerülésére inkább „maga öli meg saját magát", azaz lefagy. Nos, egy operációs rendszer esetében ez sajnos nem megoldás. Kritikus alkalmazásokra ugyanis maga a gép is csak emberi felügyelettel használható, hiszen a teljes rendszer leállása ezeknél katasztrofális következményekkel járhat.

Mit lehet tenni ilyenkor? Jobb megoldás az, amit a Novell és néhány Unix-verzió alkalmaz. Ezek a rendszerek kritikus körülmények között is képesek működni. Ha már nem tudják biztosítani valamelyik program futását, azt legalább megoldják, hogy maga az operációs rendszer minden körülmények között képes legyen futni. Hiba esetén tehát az operációs rendszer futása nem szakad meg, legfeljebb az adott alkalmazói programé. Be kell látnunk, ez elég nehéz követelmény, hiszen olyan hibák is felléphetnek, amelyeket lehetetlen rendszerszinten kezelni.

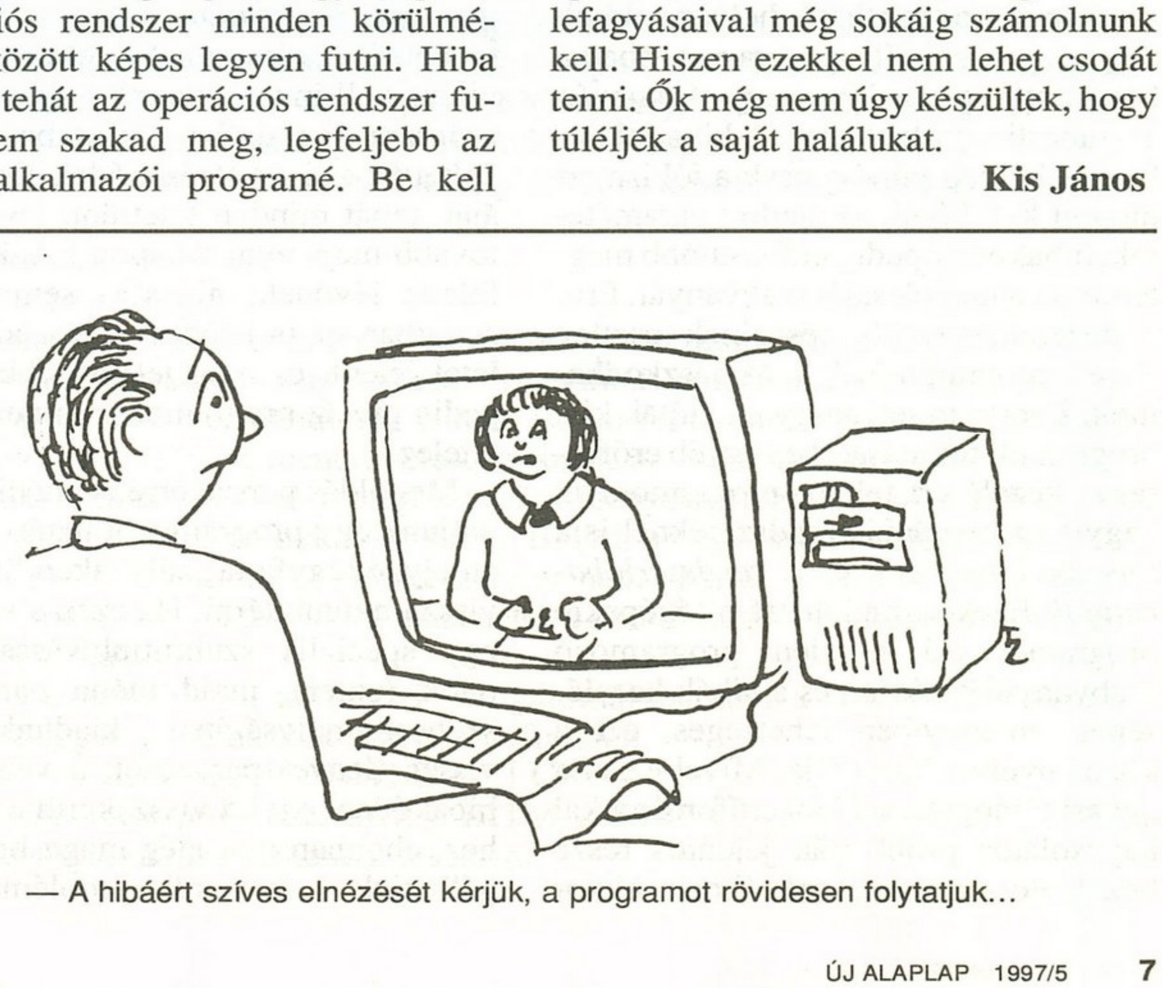
Az igazi hibatűrés

A probléma igazi megoldásához tovább kell lépni: a szoftvert és a hardvert is át kell alakítani az új filozófiának megfelelően. Ez gyakorlatilag klaszterhálózatok létrehozását jelenti (magyarul fürtözött rendszernek is nevezik). Ezeknél a rendszer megmaradt elemei bármely pillanatban képesek folytatni a megszakadt folyamatokat, mert a hardver is és a szoftver is fel van készítve a feladatok maradéktalan ellátására. Ha ez valamilyen ok folytán nem sikerül, akkor az adott hiba miatt megszakadt folyamatot újrafeldolgozással lehet pótolni.

Mindenképpen csak az ilyen, a kritikus alkalmazásokra tervezett rendszerek használhatók azokon a helyeken, ahol alapfeltétel a folyamatos, leállás nélküli működés. Az igény eredetileg a katonai alkalmazásokból származott, de mára a banki szféra, a szolgáltatások irányítása, sőt az egyre fontosabbá váló online kereskedelem és az adatbanki rendszer is hasonló követelményeket támaszt.

Hosszú út vezetett a CP/M rendszerektől a DOS-on és a Windowson keresztül azokig a ma már jól futó, klasztertechnológiát használó operációs rendszerekig és hardverelemekig, amelyeken a kritikus alkalmazások többsége fut. Ugyanakkor azt sem szabad elfelejtenünk, hogy a hagyományos operációs rendszerek és megoldások lefagyásaival még sokáig számolnunk kell. Hiszen ezekkel nem lehet csodát tenni. Ők még nem úgy készültek, hogy túléljék a saját halálukat.

Kis János



— A hibáért szíves elnézését kérjük, a programot rövidesen folytatjuk...

Az ember téved, de ott van a gép...

A hibajavítás fejlődéstörténete

A számítógép működésében előforduló hibák kijavításának feladata egyre inkább átkerül az operációs rendszerek térfelére. Amikor az ember algoritmusokban kezdett gondolkodni, azonnal felmerült az ötlet, hogy a rögzített sorrendű utasítássorozatok rendelkezzenek valamilyen hibaággal is: hibákat kezelő cselekvési tervvel, stratégiával, végső soron egy erre kitalált belső algoritmussal. Hiszen még ha csak egy egészen egyszerű problémát gondolunk is szisztematikusan végig, a döntési pontokon hamar felmerül a kérdés, hogy mi van akkor, ha itt a valóságban valami egészen más dolog fog történni, mint amire általában számítanánk.

A programozás, az operációs rendszerek és a programnyelvek fejlődése során a számítástechnikában több klasszikus hibakezelő módszer alakult ki. Ezeket — többé-kevésbé önkényesen — öt csoportba oszthatjuk. (A csoportok egyfajta időrendiséget is képviselnek.)

Hibakezelő függvény vagy szubrutin

Már egészen alacsony szinten strukturált programok írásakor is igény, hogy a hibakezelést valamilyen elkülönített, áttekinthető, és a későbbiekben is egyszerűen karbantartható helyen oldjuk meg. A procedurális programozásban a függvény, a procedúra, vagy régi nevén a szubrutin szolgál erre. A hiba jelentkezése helyén mindig csak a jól ismert modult kell hívni, az átadott paraméterek, hibakódok pedig előbb-utóbb megteremtik a maguk saját szabványát. Erre a program további részeinél, esetleg egyéb programoknál is támaszkodhatunk. Pontosan ugyanígy alakultak ki a programok futtatásáért és egyéb erőforrások kezelésért felelős programoknál, vagyis az operációs rendszereknél is a *rendszerhibakódok* és a *rendszerhibarutinok*. Ezeket az adott számítógépekre programot író minden programozó szabványnak tekinti, és a hibák kezelésénél, amennyiben lehetséges, ezt a közös nyelvet használja. Mivel ezzel a „gyári támogatással" az erőforrásokkal kapcsolatos problémák jelentős része kezelhető, a programozóknak ekkor már csak saját alkalmazásuk speciális hibareakcióira kell hibakezelő rutinokat írniuk.

Az enter-leave technika

Az előbbi módszer bonyolult programoknál csak nehézkesen használható. Mitévők legyünk például, ha egy moduláris programnál a szubrutinhívásokban egyre mélyebbre ereszkedve a hiba valahol az n-edik emelet mélységében jelentkezik? Meghívhatjuk ugyan a hibát jelző és kezelő rutint, de ha a hiba természete olyan, hogy a feladatot már nem tudjuk folytatni, akkor a „hogyan tovább?" kérdésének eldöntéséhez vissza kell innen jutnunk a felszínre, a program fő sodrába, ciklusába. Ha kiépítettük a visszatérési kódok hierarchiáját, tehát minden felettünk lévő rutin tovább tudja adni a kapott hibakódot a felette lévőnek, nincs is semmi baj. Azonban ez is jelentős futási időtöbbletet jelent, és még jelentősebb, főleg pedig precíz programozási munkát feltételez.

Megoldás persze erre is adódik. Csináljunk egy programozói határvonalat, amelyhez egyfajta „kályhaként" mindig vissza tudunk térni. Ha ezen a vonalon egy speciális szubrutinhívással átlépünk (enter), majd utána bármilyen szubrutinmélységben kiadunk egy vissza (leave) parancsot, a vezérlés a hibakóddal együtt visszakerül az enterhez, ahonnan már elég magasból vizsgálhatjuk a felmerült problémát. Ma már ez a technika is annyira elterjedt, hogy az enter-leave utasításokat néhány mikroprocesszor utasításkészletébe eleve be is építik.

Panic, exception, signal események

Ezek mindegyike valamilyen fatális, a program szempontjából kezelhetetlen hiba esetén lép fel, és mindegyikük megoldását támogatja a programot futtató számítógép, pontosabban annak operációs rendszere. Hasonlít az enter-leave technikához, hiszen a probléma felmerülése esetén egy időre, vagy akár végleg elveszíti a program a vezérlést, és az átkerül egy előre meghatározott másik pontra. Itt egy tőle független „intelligencia" (általában az operációs rendszer) próbál úrrá lenni a helyzeten. Ide tartozik a DOS-ban az *Overflow* és *Divide by zero*, a Windowsnál a *This program has performed an illegal operation* jellegű utolsó sóhaj, a Unixban a *Death of child*, a *Hangup* és a *Kill* aszinkron jelzések, valamint a Psion kéziszámítógépeknél a *Panic number 145, IPCS Message read failed* típusú rendszerüzenet. (Magyarul: túlcsordulás, nullával való osztás, ez a program illegális utasítást hajtott végre, a gyerek meghalt, kiakadás, gyilkosság, 145-ös pánik, az IPCS üzenet beolvasása nem sikerült.) A hibaüzeneteken túl ezeknek a mechanizmusoknak fő feladatuk, hogy a megbotlott vagy teljesen tönkrement programokat a rendszerből kitakarítsák, és az általuk fogva tartott erőforrásokat felszabadítsák.

Erőforrás-rendőrség

Bizonyos hibák általában a modern többfeladatos (multitasking) operációs rendszerekhez kapcsolódnak. Mindig az egyes erőforrásoknak (főleg CPU-idejének és -memóriájának) kisajátítását hivatottak megakadályozni. Nézzük először a CPU-idő védelmét. Sok feladat, program párhuzamos futtatását egy fizikai processzoron csak az idő igazságos elosztásával, nagyon apróra szeletelésével lehet biztosítani. Az ezt támogató operációs rendszereket ebből a szempontból két csoportra oszthatjuk. Az ún. preemptive ütemezővel rendel-

kezők (Unix, Psion, Windows NT) képesek a vezérlést bármikor visszavenni az aktuálisan futó programtól. Egy nem preemptive rendszer viszont tehetetlen mondjuk egy végtelen ciklusba eső programmal szemben.

Jó példák erre a Windows korábbi verziói, ahol egy program lefagyása után nem volt más lehetőség, mint az egész rendszert újraindítani. Védeni kell az egyes programokat az ellen is, hogy egymás memóriaterületére firkáljanak, és az ott lévő adatokat akár szándékosan, akár véletlenül felülírják. Érdekesség, hogy egy Psion kéziszámítógép, legyen akármilyen kicsi is, a mainframe-eknél megszokott módon védekezik ez ellen. A ludas programokat egy Panic üzenettel egyszerűen kitiltja a közösségből, majd kitakarít utána. Szemben a jó öreg Windows-zal, amely a DOS-kompatibilitás megőrzése miatt nem képes erre, hadd főjön inkább felhasználóinak a feje.

Objektumorientált hibakezelés

A procedurális programozási módszereket lassan teljesen kiszorítják az újonnan megjelenő OOP nyelvek és környezetek (a C++, a Delphi, a Java stb.). De erre az útra lép lassacskán minden operációs rendszer is (a Psion vagy a NeXT kezdettől fogva, a Windows NT Cairo csak mostanában). Bár sokak szerint ehhez a technológiához komoly teljesítményű számítástechnikai motorok kellenek, cáfolatként vizsgáljuk meg a mind operációs rendszerét, mind alkalmazásait tekintve eredendően objektumorientált Psion hibakezelését.

A hibakezelő algoritmus megpróbálja mind a négy előbbi módszer előnyét és az OOP technikát egybeötvözni. Ha az alkalmazás egy új fázisba, esetleg kritikus szakaszba ér, megkéri az *alkalmazásmenedzser* objektumot, hogy hajtson végre enter jellegű állapotváltást. A dologban részt vesz egy szintén az alkalmazás rendelkezésére álló további objektum, amely az ún. CLEANUP osztályba tartozik. Ez az enter hatására „mosodájában" egy „új szekrényt nyit a szennyesnek". A továbbiakban az alkalmazás minden újonnan létrehozott vagy felhasznált objektumot (amelyek közül a legtöbb valamilyen erőforrást is takar) használat előtt beakaszt ebbe a közös szekrénybe is, majd nyugodtan dolgozik velük tovább.

Ha a kérdéses folyamat közben hiba merül fel, a program az *alkalmazásmenedzsert* felszólítja egy leave végrehajtására. Ekkor az alkalmazás logikailag visszakerül a kérdéses szakasz elejére, de eközben a háttérben ezernyi dolog történik. Az *alkalmazásmenedzser* felszólítja a CLEANUP objektumot, hogy zárja be az aktuális mosodai szekrényt, előtte azonban vegye ki belőle egyenként az oda beakasztott objektumokat. Szólítsa fel mindegyiket, hogy legjobb tudása szerint reagáljon a felmerült problémára. Minden objektum a maga sajátos módján kezeli a felmerült hibákat, sőt végül akár meg is semmisítheti magát, ezzel engedve el minden általa reprezentált erőforrást. Ezek a hibakezelő algoritmusok objektumspecifikusak, a belőlük továbbfejlesztett objektumok természetesen örökölhetik őket, így a bennük lévő tudás nem vész el.

Mivel nem is az alkalmazás, hanem az operációs rendszer az, amely cselekvésre szólítja fel őket, maga a hibakezelés a programtól tökéletesen függetlenül tervezhető és implementálható. Az alkalmazást író azt feltételezi róluk, hogy biztosan jobban tudják, mit jelent számukra egy bizonyos hiba, és azt, hogy a hibát ők tudják a legjobban kezelni. És ez így is van nemcsak az operációs rendszer által felkínált, hanem a többi, programozó által megírt objektumnál is.

Timár István

Képtelenség...

Hogyan lehet az, hogy még a legelterjedtebb, legnépszerűbb DOS-héjprogramban, a Norton Commanderben is maradtak hibák? Pedig így van, többszöri átdolgozás és sok millió önkéntes (és önkéntelen) tesztelő tapasztalatgyűjtése ellenére is. Ha a Norton Commanderben különböző képeket akarunk megnézni (F3), előfordul, hogy a kép helyett csak egy hibaüzenet jelenik meg:

Error loading image!!

Mi lehet ennek a magyarázata? Lényegében az, hogy az NC-hez tartozó képmegjelenítő (viewer) nem mindig ismeri fel a képformátumot, nem veszi figyelembe, hogy a .BMP kiterjesztés nem határozza meg azt egyértelműen. A kiterjesztésből vett információ alapján a BITMAP.EXE program nekilát a munkának. (A megjelenítéshez ennek a programnak a meghívása azonosan történik az NC 4.0-ban és az NC 5.0-ban.) Pedig tudhatná, hogy több olyan program is van, amely saját képeit .BMP kiterjesztéssel tartja ugyan nyilván, de azok képformátuma nem azonos a Windowsban használatossal.

Más esetekben is előfordul, hogy pusztán a .BMP kiterjesztés kevés a valóságos formátum megállapításához. Néhány képkonvertáló program például az .RLE fájloknak megfelelő képfájlokat .BMP kiterjesztéssel látja el, amelyeknél pedig egyfajta tömörítésben találhatók a bitmap-képek. Az említett BITMAP.EXE program az ilyen fájlokat sem képes kibontani és megjeleníteni. Mivel esetenként a .PCX fájlok megjelenítésénél is ugyanilyen hibaüzenetet kapunk, gyanítható, hogy még a .PCX formátumok egy részét sem ismeri a szóban forgó program.

A Norton Commanderen kívül más héjprogramot is lehet használni, olyat, amely külön definiált külső programként kezeli a képmegjelenítőt. Ilyen héj például a DOS Navigator. Ha ennek adjuk meg .BMP kiterjesztés kezelésére az említett programot (lehet, hogy nem egészen illendő módon, de helytakarékosan), akkor egy további hibaüzenettel is bővülhet a paletta. A következő üzenetpáros jelentkezhet:

Error creating work file!!
Error loading image!!

Ilyen kettős üzenet akkor keletkezhet, ha írásvédett lemezről, például CD-ROM-ról szeretnénk egy képet megtekinteni. Az említett program ugyanis tudja, hogy a képet hogyan kell megjeleníteni, de nem ismeri fel a rendszer számára definiált TEMP-útvonalat. (Ennek menedzselését viszont az NC korrekt módon elvégzi.) Bár a munkafájlnak az NC alatt még nem sikerült nyomára bukkannom, úgy tűnik, hogy „normális" körülmények között a BITMAP egy swapterületet létesít a lemezen. CD-ROM esetében ez természetesen nem sikerülhet. A program azonban itt nem áll meg, így a második hibaüzenetet a nem létező fájl sikertelen olvasási kísérletére kapjuk.

A hiba kiküszöbölhető, ha a DOS Navigatornál maradva a DN.VWR fájlban a következőképpen alkalmazzuk a hívási definíciót:

GIF,BMP,PCX,WMF,DIB: BITMAP !.! C:\

Ekkor már nincs probléma, mert a BITMAP.EXE a C jelű meghajtó gyökérkönyvtárát fogja munkaterületnek tekinteni.

Simay Endre István

A pult másik oldalán

Hibakezelés — programozói szemmel

A felhasználó szeretne minél több információt kapni programjáról, különösen amikor az feldobja a talpát. (Az orvostudománynak is a kórbonctan a legfejlettebb ága!) A programozó igyekszik is megfelelelni a felé áradó sokféle kíváncsiskodásnak, de mindent ő sem láthat előre. Az alábbi cikkből egy olyan programozó töprengéseit ismerhetjük meg, aki már megpróbált minden hibalehetőséget végiggondolni.

Kap az ember egy egyszerű programozási feladatot. Floppyn érkezik egy adatállomány, a neve fix, a fájl text formátumú, egyes sorok #, mások * karakterrel kezdődnek. Egy # kezdetű sort legalább 3, de legfeljebb 12 csillagos sor követ. A feladat az, hogy készíteni kell egy fájlt (egy másik floppyra), amelyben a #-tel kezdődő sorok végéhez hozzáfűzzük az őt követő sorokat, a vezető csillag nélkül.

Minden programozó tudja, hogy ilyen és hasonló feladatok százával teremnek. Nem nehéz megoldani őket, pár óra alatt megvannak, főleg ha jól választottuk meg a programnyelvet (ami esetemben az ilyen feladatoknál az AWK szokott lenni). Eredetileg a kérés egyetlen floppyra vonatkozott, de azután már minden héten jön egy, és hetente le kell futtatni a programot. Ezt persze nem a programozó végzi, hanem valaki más, akinek van némi PC-s tapasztalata, de nem túl sok.

Látszólag egyszerű

Mit csinál a program? Beolvassa a fájlt a merevlemezre, hiszen csak egy floppyegység van. Közben összefűzi a rekordokat. Ekkor beteszik az output floppyt, és a program kiírja rá a kész fájlt. Ha csak ennyi volna a feladat — és sajnos sokan azt hiszik, hogy ennyi —, könnyű dolga volna a programozónak.

Először is itt van a floppy. Hogy hívják az input fájlt? Az eredeti feladatban meg volt adva a neve. Legközelebb is biztos az lesz, de mi van, ha mégsem. Tehát fel kell ajánlani az alapértelmezett nevet, viszont lehetőséget kell adni, hogy aki futtatja a programot, az ezt felülírhassa. És mi legyen, ha megad egy másik nevet, de nincs olyan nevű fájl a lemezen? Fel lehet kínálni valamilyen könyvtárlistát, hogy abból válasszanak, vagy adjon a program ilyenkor egy hibaüzenetet, és leállhat.

És ha másképp van?

Jó, tegyük fel, hogy sikerült megfelelő fájlnevet kapni, jöhet a beolvasás. És mit tegyünk, ha nem #-tel kezdődik az első sor? Rendben van, adjunk egy hibaüzenetet. Ez még nem elég egyértelmű: legyen mondjuk *adattartalom hiba*. Esetleg a hiba mellé kiírhatjuk annak a sornak a számát is, ahol a hibát észleltük. Aki futtatja a programot, az ennyiből megértheti, hogy baj van az állománnyal, szólhat, ha csinálni kell valamit.

Hasonló a gond és a teendő, ha a két megengedett kezdőkarakteren kívül egy harmadik fajtát észlelünk. Szinte ugyanez a helyzet, ha a megengedett háromnál kevesebb, vagy 12-nél több csillagos rekord követi a fejrekordot.

Ekkor már felrémlik egy általánosabb hibakezelő rutin szükségessége. Ez tájékoztathatna arról, hogy hányadik sorban milyen típusú hiba van. Esetleg még azt is közölhetné, hogy hányadik #-es sor után vagyunk éppen — ez megkönnyítené az állományt generáló program és programozója dolgát. Hiszen valahol, valamikor a mi input anyagunk is output volt.

De mi legyen utána? Próbáljuk-e továbbolvasni a rekordokat, vagy álljunk meg? Ez már a program megrendelőjén múlik. Ő tudja, hogy egy részleges output szükséges-e, vagy éppenséggel bajt okozna.

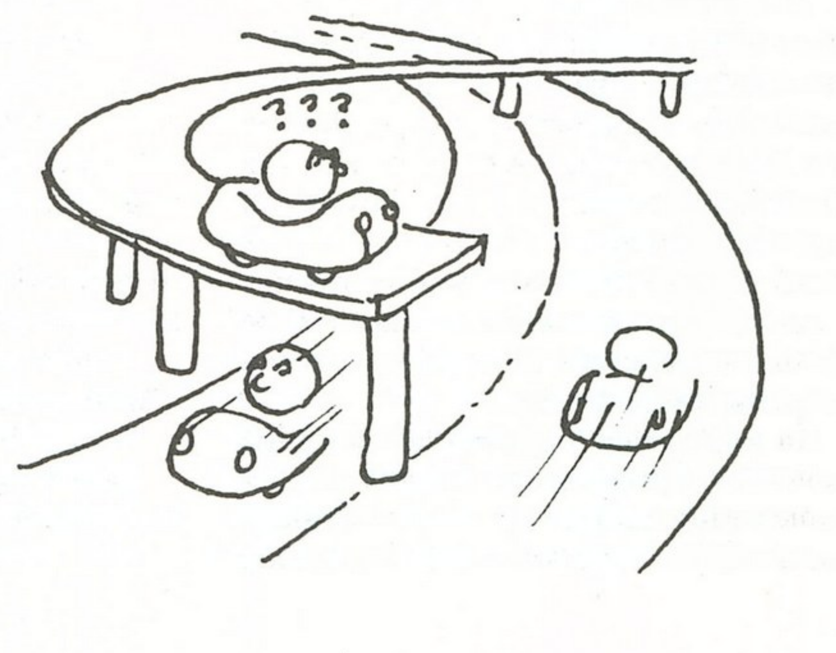
Mennyit vegyünk a nyakunkba?

Tegyük fel, hogy az előbb említett hibák egyike sem lépett fel. Még mindig csak az inputnál tartunk, és van további két kellemetlen eset.

Az egyik az, ha a floppy fizikailag hibás. Előfordulhat ez a beolvasás elején is, vagy közben valahol, a mi szempontunkból mindegy. Ekkor általában átkerül a vezérlés az operációs rendszer szintjére. Ez elkapható, bár nem minden programnyelvben egyszerű. Ráadásul olyan angol hibaüzenetek jönnek ki, hogy ha a programot kezelő személy nincs rá felkészítve, megrémülhet tőlük. Ha viszont mi akarjuk az ilyen hibákat kezelni, komoly munkát veszünk a nyakunkba. Nagy kérdés, hogy megér-e ennyit egy ilyen feladat.

A másik kellemetlen eset, ha az adott fájl véletlenül nem szöveges, például valaki egy bináris fájlt helyezett el az általunk várt névvel a floppyn. Ilyenkor a beolvasás meg tud bolondulni, például mert várja a sorvége jelet, ami nem jön. Még az is megtörténhet, hogy elszáll a program, mert túl hosszúnak érzi a sort. Vagy olyan karaktert talál benne,

PATH NOT FOUND



amellyel a sorolvasó nem tud mit kezdeni. Tehát elvben ilyen esetekre is fel kell készülni, de ennek a kezelése már tényleg komoly munka.

Ha a fenti — egyébként tényleg ritka — rémségek egyike sem következett be, akkor simán beolvassuk és rámásoljuk a fájlt a merevlemezre. De mi történik, ha betelik a lemez, vagy ha fizikai hiba van az írásnál? A betelésre rendszerint kezelhető hibát kapunk az író rutintól, erre elég könnyen fel lehet készülni. Az íráshiba pedig hasonló problémákat vet fel, mint a floppy olvasási hibája.

A kiírás nem tartalmaz sok elvi újdonságot. Esetleg kellemetlen lehet, ezért fel kell rá készülni, hogy nem a megfelelő floppyt tették be. Ezt valószínűleg nem tudjuk ellenőrizni, hacsak nem végzünk a lemezcímkére vonatkozó ellenőrzést. Ennek programozása és menedzselése megintcsak bonyolult feladat. Baj lehet még, mondjuk, hogy kiírás közben valaki kiveszi a floppyt — kemény rendszerüzenet vagy programelszállás lesz a következménye. Az biztos, hogy megéri a programba egy olyan opciót is beépíteni, hogy egy már korábban beolvasott és ellenőrzött fájl tartalmát írja ki floppyra. Ez főleg akkor hasznos, ha aránytalanul hosszú és bonyolult a beolvasó és feldolgozó rész.

A fentiek mind az egyszerű, egyfelhasználós, egyfeladatos környezet (nem is teljes) végiggondolásából származnak. A lehetséges problémák száma és bonyolultsága aránytalanul megnő, ha a készítendő program egy nagyobb operációs rendszer egyik taszkja lesz. Igaz viszont az is, hogy az ilyen rendszerek eleve jobb hibamegelőző és hibakezelő eszközöket adnak (lockolás stb.), melyeket fel lehet hozzá használni.

Mindenre felkészülni

Gondoljuk csak el, kiindultunk egy mindennapos, egyszerű feladatból, amely elvileg „simán lekenhető", és kiderült, hogy korrekt elintézése bonyolult rutinok írását, a rendszer olykor egészen mély ismeretét tételezi fel. A fenti hibaesetek ugyan nagyon ritkák, de ha egy programot kiadunk a kezünkből, és nem akarjuk időnket a panaszkodó felhasználók megnyugtatására, vagy a több évvel ezelőtt megírt programok esetleg meg sem lévő forráskódjának böngészgetésére fordítani, akkor nem árt, ha komolyan vesszük a valószínűtlen, de nem kizárt hibák lehetőségét is, és legalább elvben felkészülünk rájuk.

Horlai János

Önismereti teszt

Nem mindegy, melyik DOS

A rendszer verziószámának hibás felismerése számos bonyodalom forrása lehet. Hogy ez a látszólag egyszerű feladat is magában hordozza a tévedés lehetőségét, azt igyekszik bemutatni (és orvosolni) ez a cikk.

Ha a gépünkön futó operációs rendszer verziószámára lennénk kíváncsiak, MS-DOS és vele kompatibilis operációs rendszerek alatt adódik a lehetőség: begépeljük a VER utasítást, és a gép megmondja. De. Az MS-DOS 5.0-ban és az utána következő változatokban megtörténhet, hogy a valóságostól eltérő verziószámról kapunk üzenetet. Hasonlóan téves eredményt kaphatunk akkor is, ha programból szeretnénk detektálni a DOS éppen futó verzióját. Pedig még csak az MS-DOS berkeiben keresgélünk! Mi minden történhet a többi DOS-típussal és azok verzióival?

Két olyant is találhatunk a DOS megszakításai között, amelyet felhasználhatunk erre a célra. Az első már az MS-DOS 2.0 verziójától létezik, és megtalálható az e rendszerrel kompatibilis operációs rendszerekben is. Ez a szélesebb körben használt megoldás a 21h DOS-megszakítás 30h szolgáltatását használja, és visszaadja a fő- és alváltozatszámot. Hátránya viszont, hogy e szolgáltatással adott válasz az említettek szerint befolyásolható (SETVER.EXE).

A Ralf Brown-féle megszakításlista áttekintésével azonban adódik egy másik út is. Ez az MS-DOS 5.0 óta létező lehetőség a 21h 3306h szolgáltatása, amely a valós verziószámot adja vissza. A probléma most már csak az eredmények értelmezése. Néhány esetben azonban az alap operációs reendszerre is következtethetünk a visszaadott verziószámból. Így a Win95 MS-DOS v7.0-nak tekinti magát egy DOS-ablakban (21h/3000h), míg a Windows NT MS-DOS v5.5-nek (21h/3306h). Ha a verziószám 10.0 vagy e feletti, akkor pedig valószínűleg az OS/2 valamelyik változata van a gépen. Néhány más esetben azonban a verzió ilyetén lekérdezése még MS-DOS-kompatibilis rendszerben sem minden esetben pontos. Például az IBM DOS 6.1 6.0-nak azonosítja magát, a DR-DOS 5.0 és 6.0 pedig 3.31-nek.

Ilyenkor segíthet az, hogy a 21h/3000h szolgáltatással lekérdezhetjük a DOS gyártóazonosítóját (OEM-szám). A Digital Research (DR) termékei esetében pedig lehetőség van egy másik szolgáltatással a gyártmányazonosító lekérdezésére. Ezzel meglehetősen pontos meghatározásra van lehetőség. A DOS megszakításain keresztül néhány más operációs rendszer szintén beazonosítható. Ilyenek a FreeDOS, a WinDOS, a PCMOS vagy a VMIX2.

A fentiek alapján egységes eljárás írható az operációs rendszer, illetve a verzió lekérdezésére. Ennek egyik Pascal megvalósítása a lemezmellékleten található. A eljáráshoz definiálásra került egy WORD hosszúságú Record a verziószámoknak, illetve egy külön típus a különböző operációs rendszereknek. Ezzel lehetőség nyílt, hogy a DOS, illetve a Windows verziói, illetve az operációs rendszer számára önálló változókat hozzunk létre, melyeket az eljárás feltölt értékekkel. Így egyszeri eljáráshívás után globálisan rendelkezésre állnak a fenti adatok.

Az eljárás lehetőség szerint teszteli néhány más program jelenlétét is. Ilyen a DesQview, a Volkov Commander, 4DOS, NDOS, illetve a DOS Navigator, mely utóbbi jelenlétének tesztelése a fejlesztő által publikált kód felhasználásával történik. Ezek az adatok, akárcsak a DOS memóriabeli elhelyezkedése, logikai változókon keresztül érhetők el.

A DOS_VERSION eljárás a jelen formájában keverten alkalmazza a tiszta Pascal, illetve az abba integrálható Assembler lehetőségeit. Ha nem is túlságosan elegáns ez a megoldás, tapasztalataim szerint elég pontosan működik, és további ismeretek birtokában mindenki szabadon átfogalmazhatja.

Simay Endre István

A szakadék szélén táncolva

Hibatűrő programok készítése

Vajon meg kell-e várnunk, míg beleszédülünk a szakadékba? Nem lehet-e már akkor is védekezni a hibák ellen, amikor még nem következtek be? Cikkünk az idejében megtett intézkedések lehetőségét vizsgálja, amikor a futó program az utolsó pillanatban még kézben tartja az események irányítását.

Minden programozó ki van téve különböző kellemetlen meglepetéseknek. Mondjuk annak, hogy programja futás közben hirtelen leáll egy általános védelmi hibával. Vagy túlcsordul a program rendelkezésére álló memória, vagy negatív értéket visz magával az a változó, amelyet átadott gyökvonásra. Teljesen mindegy, hogy a hardver vagy a szoftver hibájából történik-e a baj, tény az, hogy a rendszer — különösen windowsos környezetben — könnyen fejreállhat.

Ezek a veszedelmek nem csak a programozóknak okoznak fejfájást. *A rendszerfejlesztők is régóta igyekeznek minél robusztusabbá, minél hibatűrőbbé tenni rendszereiket,* hogy a felhasználó lehetőleg ne is értesüljön a kisebb, menet közben elhárítható hibákról.

Legalább szólna...

Amikor a felhasználó azt érzékeli, hogy lemerevedik a rendszere, akkor már régen rossz. Se té, se tova. De kénytelenek vagyunk-e bevárni a legrosszabbat? Azt, hogy az operációs rendszer a végső eszközhöz folyamodjon, és kimondja a végítéletet: állj! Nem lehetne-e idejében észrevenni és jelezni *a hibaállapotok várható bekövetkezését* a futtatott felhasználói programból, és megtenni a kellő védőintézkedéseket? Nem hagyni, hogy az operációs rendszer vegye át a vezérlést, mert akkor már aligha enged vissza minket a kormányhoz.

Persze az operációs rendszertől is elvárható lenne, hogy ha csak lehet, és ameddig lehet, ne leállással, hanem másképpen hozza tudomásunkra a bajt. Vagy ha már a lemerevedést nem tudja megakadályozni, legalább azt tegye meg, hogy *a leállás után elegendő információt bocsásson a programozó rendelkezésére,* aki így kibogozhatja, mi is történt. És végezze el a rendszer automatikusan az „elsősegélynyújtást", például *zárja le, mentse el a nyitott állományokat.* A legjobb persze az lenne, ha megmaradhatna a vezérlés a futtatott alkalmazói programnál. Vagy visszakerülhetne oda. Az operációs rendszer, ha tud, a hibák előrejelzésére nyújtson támogatást, és ne lökje félre a volántól a felhasználót.

Kívülről jön a veszély!

Érezhető, hogy a probléma gyökere a futás közben előforduló hibák kezelése körül van: az a kérdés, hogy ezek megközelíthetők-e valahogy másképpen és máshonnan. Nem lehetne-e például magukat a programokat felkészíteni arra, hogy a váratlan események, a szoftver- vagy hardverhibák, vagy a nem is olyan ritkán előforduló kezelési ügyetlenségek ne járjanak végzetes következményekkel?

Minden kívülről fenyegető veszélyt kivédeni nyilván nem lehet, főleg nem egyedileg, minden programban különkülön. De talán nem is lenne érdemes. Egyrészt óriási mértékben megnövelné a programok terjedelmét, másrészt rengeteg kód megismétlődését jelentené, hiszen *a külső fenyegetések többsége minden programnál azonos.* Még akkor is így van ez — legalább megjelenési formájában —, ha a hiba okozója maga a programozó. (Aki például nem mérlegelte elég gondosan a helyzetet, mikor beállította az indexhatárokat. Vagy nem gondolt a nullával való osztás előfordulásának lehetőségére.) Dehát mért legyen a lehetséges hibák végiggondolásának visszatérő feladata kizárólag a programozó dolga? A fájlok megnyitásának, lezárásának bonyolult procedúrájával sem nekünk kell bíbelődni. Miért nem nyújt több segítséget az operációs rendszer a hibák elhárításához futás közben is?

Más bonyodalmak is vannak azzal, ha menet közben akarjuk a programból figyeltetni a külső veszedelmeket. Egy futó program (vagy rutin) a közvetlenül őt aktivizáló környezetet még „látja", az esetek nagy részében azonban hagymahéjszerűen rétegeződnek egymásra a különböző rutinok, programok. Megvalósítható ugyan, hogy az előforduló hibáról több rétegen át oda-vissza menjen az üzenet, de az információ továbbítása nem egyszerű feladat. (Ráadásul nem is veszélytelen, mert *az előforduló hiba a program környezetében is változásokat okozhat.*)

A hiba mint kivétel

A „robusztus", hibatűrő programok készítésének legelegánsabb módja az, ha maga a programnyelv ad lehetőséget olyan védelmi apparátus aktivizálására, amely segít a futási hibák idejében történő előrejelzésére, és amennyiben lehetséges, ezek elhárítására. Már az Ada nyelv követelményeinek megfogalmazásakor gondoltak arra, hogy opcionálisan be kellene építeni a programokba valamilyen kivételkezelő mechanizmust, amelynek elsődleges felhasználási területe éppen a futás közben előforduló hibák kezelése lenne.

A programnyelvek általános követelményévé ez akkor kezdett válni, amikor az objektumorientált programozás széltében-hosszában elterjedt, hiszen ez kiváló eszköz az anomáliák elleni védekezésre is. A C++ nyelv továbbfejlesztésével foglalkozó munkacsoport már a szabványosításról is gondoskodott: az ISO WG21/ANSI X3J16 ajánlás alapján készülő rendszerek olyan kivételkezeléssel rendelkeznek, amely hordozhatóságban messze felülmúlja elődeit, valamint enyhíti azok típusbeli korlátozásait. Az MFC — Microsoft Foundation Class — könyvtár 3.0 változata is eleget tesz a C++ iránt támasztott újabb követelményeknek. Többek között a Visual C++ új verziója is ezt az igényesebb megoldást alkalmazza. (Nem árt tudnunk, hogy a Windows NT és a Win95 még olyan — makrókra

épült — kivételkezelési technikával végezte a C nyelven írt kódok feldolgozását, amelyeknél a típus csak UNSIGNED INT lehetett, ráadásul C++ objektumok esetén a rendszer időnként hibás destruktorhívást produkált.)

A C nyelv azonban aligha lesz valaha is a hétköznapok programnyelve. Profik kezében kitűnő eszköz, de annyira mindent megenged, hogy az általa nyújtott túlzott szabadság sok esetben veszélyes lehet. A programfejlesztés sokkal biztonságosabban kézben tartható egy Pascal-stílusú nyelvvel, különösen, ha olyan kellemes és felhasználóbarát keretrendszer áll rendelkezésünkre, amilyet a Turbo Pascal IDE integrált fejlesztő környezete biztosít.

Komponensalapú Pascal

A Pascal nyelv fejlődése az utóbbi időkben újabb lendületet vett. Kettős cél elérése lebegett a fejlesztők szeme előtt: a nyelv erejének és a fejlesztő-környezetnek a megújítása. A nyelv további kiterjesztésével nem kevesebbet akartak elérni, mint hogy Pascal-szerű nyelven is a C/C++ nyelven írottakkal közel azonos hatékonyságú programokat lehessen készíteni. Ugyanakkor a Pascal hagyományaihoz való ragaszkodás is teljesen érthető, hiszen az sokak tapasztalata szerint könnyebben írható és tesztelhető, és nem utolsósorban áttekinthetőbb programokat eredményez. A fejlesztőkörnyezet továbbfejlesztését főként az igények fokozódása indokolta: mindenben ki akarták használni a vizuális programfejlesztés lehetőségeit, átmentve azonban az új környezetbe mindazt a kényelmet, amit az IDE nyújtott. Egyszersmind azt is számításba vették, hogy a vizuális technika új lehetőségek beépítését és könnyű kezelését is lehetővé teszi.

A nyelv bővítése gyakorlatilag már az Object Pascal létrehozásakor megtörtént. Az Object Pascal utódjaként a Delphi azzal koronázta meg a fejlesztést, hogy uralkodóvá tette a grafikus képernyőt, megteremtette a grafikus felületen működő Debuggert, és a felhasználó rendelkezésére bocsátott egy olyan vizuálisan kezelhető könyvtárat, amely maximális mértékben támogatja az igényes és gyors programfejlesztést.

Külsőre az új rendszerben írott programok alig különböznek a Borland Pascalban megírtaktól, valójában azonban már az Object Pascal jelentős előrelépést hozott az egész OOP koncepció történetében. A problémák megoldásához itt többre volt szükség, mint az

Rokoni munkamegosztás

1. Közvetlen leszármazottak

EIntError	Az egész számokkal végzett műveletek hibáit felölelő kivételosztályok közös őse. Ennek lehet előírni a közös tennivalókat.
EMathError	A lebegőpontos aritmetikai kivételosztályok közös őse.
EProcessorException	A hardverkivételek őse.
EOutOfMemory	Arra jó, hogy szükség esetén a programozó gondoskodni tudjon például a dinamikus helyfoglalásról.
EInOutError	Meg lehet nézni az ErrorCode mezőt.
EInvalidParameter	Ellenőrizni lehet, hogy mi a hiba egy memóriarész felszabadítását kérő pointerrel.
EInvalidCast	Miért nem egyezik az AS operátor bal oldalán megadott objektumosztály a jobb oldalán megadott objektumosztállyal.
EConvertError	Miért nem tudja a konverziós függvény elvégezni az előírt átalakítást.
EAbort	Lehetőséget ad arra, hogy dialógusablak nélkül, csendben kilépjünk.

2. Unokák

A) Egész aritmetikai kivételek, mindegyik közvetlen leszármazottja az EIntErrornak.

EDivByZero	Nullával való osztás.
ERangeError	Hibásan megadott intervallum — akkor derülhet ki, ha értéktartomány-ellenőrzést kértünk a megfelelő fordítási direktíva segítségével a kijelölt programrészre.
EIntOverflow	Túlcsordulási hiba — akkor észlelhetjük, ha a megfelelő fordítási direktívával ellenőrzést kértünk.

B) Hasonló kivételosztályok vannak a lebegőpontos aritmetikára is. De az EInvalidOp kivételosztály megjelenéséből nem csak arról értesülünk, hogy egy érvénytelen matematikai művelet végrehajtásával próbálkoztunk, hanem arról is, hogy az operandussal van a baj, például negatív számból akartunk négyzetgyököt vonni.

C) EFault. A processzorhiba-kivételek őse a hardverkivételek osztályán belül. Ennek gyerekei az előre definiált dédunokák.

3. Dédunokák

EGPFault	Az alkalmazás érvénytelen memóriacímeket akar elérni. Gyakran az az oka, hogy nincs inicializálva a kívánt objektum, vagy olyan pointert akartunk megsemmisíteni, amelynek inicializálásáról elfeledkeztünk.
EPageFault	A CPU laphibát észlelt.
EInvalidOpCode	A CPU érvénytelen utasításra bukkant.
EBreakpoint	A CPU töréspontot talált.
ESingleStep	A CPU single-step üzemmódban vár a következő tennivalóra.
EStackFault	Nincs elég hely a stackben az eljárás vagy a függvény lokális változóinak elhelyezésére. Akkor értesülhetünk a stack túlcsordulásáról, ha fordításnál külön kérjük az ellenőrzést a megfelelő fordítási direktíva segítségével.

objektumorientált programozás módszerének sima alkalmazására — valójában új elvek kikristályosodásának lehettünk tanúi. Az Object Pascalban (amely már a későbbi Delphi igényeinek kielégítésére készült) központi szerepe lett azoknak a különleges objektumoknak, amelyek tervezés közben és futási időben egyaránt elérhetők. Az ilyen objektumokat hívja az Object Pascal *komponenseknek* vagy *alkotóelemeknek.*

Gondoljunk csak bele: az objektumok megjelenése eltüntette azt az éles különbséget, amely a programok kódját és adatait szétválasztotta. Most a komponensek megjelenésével megszüntethető a programtervezés és a futtatás merev szétválasztása. A komponensek tulajdonságai tervezéskor is, futási időben is láthatóvá tehetők, kiegészíthetők, módosíthatók. Még a hozzáférés módjában is megvalósítható, hogy tervezési időben alakítsuk a futási időben érvényesülő tulajdonságokat, kiválasssszuk a megfelelő komponenseket, hozzájuk rendeljünk újabb tulajdonságokat, vagy megváltoztassuk az értéküket.

Vészhelyzet

A beépítésre váró újabb lehetőségek között előkelő helyen állt azoknak a rendkívüli eseményeknek a programból való kezelése, amelyekről hagyományos módon csak a futás közben előforduló katasztrófákból szerezhetünk tudomást. A feladat itt valami olyan szoftvereszköz kifejlesztése volt, amely idejében elfogja azokat a hibaállapotokat, amelyek egyébként az alkalmazói programok leállásához vezetnének, hogy átalakíthassa őket speciális módon kezelendő eseményekké. Elvileg ez nem különbözik a C++ számára definiált kivételkezeléstől, a megvalósításhoz azonban (1) végre kellett hajtani a nyelvben a szükséges bővítéseket, (2) ki kellett dolgozni az Exception osztály hierarchikusan felépített rendszerét, (3) ki kellett fejleszteni hozzá egy rendkívül hatékonyan működő rutinegyüttest.

Az újonnan kifejlesztett eszközöket egyetlen új modulba építették bele, a SysUtilsba, amely INCLUDE paranccsal UNIT-ként hozzáilleszthető a felhasználói programokhoz. Ebben található az Assembly szinten kifejlesztett rutinok együttese, valamint a fenyegető veszedelmek felderítésére szolgáló kivételosztályok jól kiépített hierarchiája. A SysUtils objektumosztályai és rutinjai azzal, hogy az Object Pascal futásidejű könyvtárához épülnek hozzá, a futásidejű HIBÁKAT automatikusan KIVÉTELEKKÉ tudják átalakítani. Ha olyanok a kivételek, hogy azokkal maguk is el tudnak bánni, megteszik, ha nem, akkor jelzést adnak, és a programozó számára lehetővé teszik a tetszőleges eseményfüggő tevékenység elvégeztetését. Sok esetben ez azt jelentheti, hogy valamilyen stabil helyzetből egyszerűen újraindul a program, vagy egy hibakezelő rutin lép működésbe.

A kivételek rendszere

A mellékelt keretes anyagban láthatjuk, hogy milyen kivételosztályok kezelésére van előre felkészülve a SysUtils. Közvetve vagy közvetlenül valamennyi leszármazottja a legáltalánosabb ősnek, az Exception osztálynak. Az előre definiált kivételosztályokat a felhasználó természetesen saját szájaíze szerint további osztályok definiálásával egészítheti ki.

Példaként nézzük meg, hogyan lehet Delphi nyelven egy olyan FILECOPY.PAS programot készíteni, amely biztonságosan, a kivételosztályok figyelésével másol egy adatfolyamot. (A példát a „Delphi — út a jövőbe" c. könyvből vettük, melynek ismertetése 1996/6. számunkban olvasható. Néhány bonyolultabb példával együtt ez a példa is megtalálható a lemezmellékleten.)

Legyen a program neve is Filecopy. A tevékenység végrehajtására definiálnunk kell egy függvényt, amelynek meghívása lényegében a programtörzs egyetlen tennivalója. A Copyit(honnan,hova) kétváltozós függvény argumentumaiként adjuk meg a forrásfájlt és a célfájlt.

A programtörzsben TRY kulcsszóval kell megjelölnünk, hogy honnan kezdődjék a figyelés. Mindaddig, amíg minden rendben megy, az utasítások normális módon, sorban végrehajtódnak. Mihelyst azonban valami baj van, a vezérlés átadódik az EXCEPT kulcsszó utáni résznek. Ez a programrész, az EXCEPT-től az END-ig akkor és csak akkor kerül végrehajtásra, ha az Exception alapján definiált kivételosztályok valamilyen baljós eseménye következik be.

Üzenetek beépítése a programba

Mi történik akkor, ha a programtörzsbe a Copyit eljárás hívása elé beírjuk a TRY kulcsszót, utána pedig egy olyan sort, amely a sikeres másolásról ad hírt, majd az EXCEPT kulcsszó után, de még az END elé egy sikertelenségről szóló üzenetet? Feltéve, hogy a Copyit függvény már megfelelő módon definiálva van, két eset lehetséges. Ha a függvény végrehajtása sikertelen, akkor a hívásról közvetlenül az EXCEPT utáni részre kerül a vezérlés, és megjelenik a képernyőn az üzenet a sikertelenségről. Ha sikeres volt a hívás, akkor viszont a hívást követő üzenetküldést végzi el a rendszer, amely a sikeres másolásról ad hírt, és kimarad a hiba esetén elvégzendő tevékenység.

Mint látható, a hibakezelésnek a programból való vezérlése nem kívánja azt, hogy a normális programlogikába belekutyuljuk a hibakezelési logikát. Ha akarjuk, egyszerűen megtarthatjuk azokat hibakezelési eljárásokat, amelyek a SysUtilsba vannak beépítve, ha többet vagy mást akarunk, akkor pedig magunk kezelhetjük a rendkívüli eseményeket a rendelkezésünkre bocsátott nyelvi apparátus és a szokásos objektumkezelési módszerek segítségével.

Bonyolultabb esetek

Egy másik példán láthatjuk, hogy a TRY blokkokat egymásba skatulyázva is lehet alkalmazni. A lemezmellékleten közölt példákon azt is megfigyelhetjük, hogy az alapértelmezés szerinti hibakezelés megvalósításához még egyszerűbben járhatunk el. Az EXCEPT kulcsszó helyett a FINALLY kulcsszót adva meg, csak annyit kell írnunk, hogy mi legyen az a normális tevékenység, amelyet tetszőleges (de természetesen a kivételosztályokban definiált!) hiba jelentkezése után el kell végeznünk. (Például a fájl lezárása.)

További kulcsszó segítségével az is megvalósítható, hogy a hiba felléptekor a kivételobjektum birtoklási jogát átadjuk a kivételkezelő logikának. Erre szolgál a RAISE utasítás, amelynek argumentumában azonban objektumot, nem pedig osztályt kell megadnunk. A kivételobjektumot ilyenkor közvetlenül az argumentumban hozhatjuk létre oly módon, hogy meghívjuk a megfelelő kivételosztály CREATE konstruktorát. A kivételkezelés elvégzése után a kivételobjektum automatikusan törlődik (a DESTROY destruktor hívása révén). A RAISE utasítás a vezérlést a legbelső olyan kivételkezelőnek adja át, amely alkalmas az adott osztályú kivételek kezelésére. A RAISE utasítás használata szintén megfigyelhető a lemezmellékleten közölt egyik példán. (A példák mind a fentebb említett könyv lemezmellékletéről vannak átvéve. A programokat Juhász Mihály, a könyv egyik társszerzője készítette.)

Vargha Dénes

Mit csináljuk, ha...

Error futásidőben

Egy-egy program elindítása után, ha nincs gond, minden további nélkül élvezhetjük a futtatás eredményét. De ha van!? Akkor lehet keresgélni a hiba forrását az operációs rendszerben, a kezelői felületen, vagy magában az alkalmazói programban.

A leggyakoribb hibaüzenetek a futtatott programtól többé-kevésbé függetlenek. Ezeket az operációs rendszer produkálja, vagy esetleg az arra telepített komplex héjprogram, mint például a Windows valamelyik 16 bites verziója. E hibaüzenetek többségének föllelhető a dokumentációja, a képernyőn megjelenő üzenetek tehát a kézikönyvek birtokában megfejthetők. De hogy mit kell tennünk ilyenkor, arra többnyire még csak utalást sem találunk.

Van a hibaüzeneteknek egy még több bosszúságot okozó csoportja. Ezeket az illető program maga váltja ki, és jó esetben a program fejlesztőkörnyezetének dokumentációja is ejt róluk néhány szót. Ezek a program ún. futásidejű hibái. A program vagy a futtatókörnyezet leírásában azonban legtöbbször hiába keressük a hibák okának vagy megszüntetési módjának ismertetését.

Az e tárgykörbe tartozó üzenetek közül tekintsük át példaként — már csak elterjedtsége okán is — a Turbo Pascal környezetben fejlesztett programok néhány jellemző üzenetét. Ezekből a felhasználó gyakran csak ilyesmit lát a monitoron:

```
Runtime error 103 at 0073:0182.
```

Itt a *Runtime error* szöveget követő szám a hibakódra utal. Utána a programmodul következik, ahol a hiba jelentkezett. A legtöbb ilyen hibát a program írójának tulajdonképpen illene kezelnie — fejlesztéskor, a programba beépített kódolással, de ez gyakran sajnos nem történik meg. Viszont magát a hibakódot a Turbo Pascal fejlesztőrendszer súgója alapján azonosítani lehet, indokolt tehát nagyító alá vennünk őket.

A teljesség igénye nélkül sorra átnézzük azokat a futás közben gyakran előforduló hibákat, amelyekről minden felhasználónak érdemes többet tudnia, és rövid megjegyzéseket fűzünk hozzájuk: mit tegyünk, ha találkozunk velük. Mindegyikük kijelzésére nem is igen lehetne felkészíteni a programokat, vagy csak a kód indokolatlan felduzzasztása árán. Nyilván nem térhetünk ki a bonyolultabb egyedi esetek tarka változatosságának magyarázatára sem, hiszen ezekhez a korrekt üzeneteket legfeljebb a program szerzője tudná megfogalmazni, a konkrét helyzet ismeretében.

Lássuk először azokat az üzeneteket, amelyeknél legalább sejteni lehet, hogy mit is kell tennünk. A bővebb hibaüzenet-listát keretes anyagunk tartalmazza. Ebben megtalálhatók azok az üzenetek is, amelyeknek felbukkanására jobbára csak alaposabb elemzéssel lehet magyarázatot találni.

`002 = Fájl nem található`

A program által megnyitni kívánt fájl nem fordul elő. A hiba jelentkezésekor a program dokumentációja alapján győződjünk meg róla, hogy nem töröltünk-e ki valamit véletlenül.

`003 = Hibás elérési útvonal`

A program által keresett elérési útvonal hibás. Leggyakoribb oka, hogy a program belső konstansként kezel egy elérési útvonalat, és azt megváltoztattuk. Például más könyvtárnevet adtunk meg a program telepítésekor, mint amit a program kínált, vagy időközben megváltoztattuk a könyvtár nevét. A megoldás kulcsa szintén a programdokumentációval végzett egyeztetés lehet.

`004 = Túl sok nyitott fájl`

A program futás közben újabb fájlt kíván megnyitni, de az operációs rendszer nem támogatja további fájlok nyitását. Megoldás lehet a DOS CONFIG.SYS állományában a FILES=xx értékének növelése.

`005 = Fájlhozzáférési hiba`

Oka legtöbbször, hogy a program *read only* attribútumú fájlba szeretne írni. Ez olyan esetekben szokott előfordulni, amikor például CD-ről másolunk fájlokat, és a másolóprogram írásvédelemmel látja el a másolt állományokat. Adjuk ki a DOS-ban az ATTRIB –R *.* parancsot.

`101 = Lemezírási hiba`

Előfordulásakor a munkalemez betelésére gyanakodhatunk. Legtöbbször akkor fordul elő, ha floppyról indítunk egy egyébként merevlemezre szánt és ahhoz paraméterezett programot.

`103 = A fájl nincs megnyitva`

Logikusan a hibaüzenet akkor fordul elő, ha meg nem nyitott fájlból akarunk olvasni. Gyakran azonban olyankor is megjelenik, amikor a fájl nyilvánvalóan megnyitható, például éppen a futó program állománya. Ilyenkor bonyolultabb lehet a helyzet. A program írója esetleg azt a lehetőséget akarta kihasználni, hogy az .EXE fájl végére overlay és más adatállomány is függeszthető. Ugyanezt a hibaüzenetet kapjuk, ha a program ilyeneket használ, de az olvasni kívánt .EXE fájl *read only* jellegűre van beállítva. Önálló .OVR fájl esetében a *read only* attribútum általában nem okoz galibát. A megoldás azonos a 005-ös hibánál leírtakkal.

`150 = A lemez írásvédett`

A lemez, amelyen a program dolgozni szeretne, írásvédett. Ez a hiba általában floppyról indított, és azon munkaállományt létrehozó program futtatásakor szokott jelentkezni. Oldjuk fel az írásvédelmet, vagy indítsuk merevlemezről a programot.

`152 = A meghajtó nem áll rendelkezésre`

Leggyakrabban akkor fordul elő, ha floppyról indítottuk a programot, az dolgozna is rajta, de közben kivettük a lemezt a meghajtóból. Előfordulhat, hogy elfelejtettünk lemezt tenni a meghajtóba, amikor merevlemezről elindítottunk egy archiváló funkciójú programot.

157 = Ismeretlen hordozótípus

Ugyancsak az előzőekben ismertetett körülmények fordulhatnak elő, ha felismerhetetlen floppyt helyeztünk a meghajtóba. Lehet, hogy a lemez más operációs rendszerhez van formattálva, illetve formattálatlan. A lemez 0. szektorának sérülése egyébként ugyanilyen hibát eredményezhet.

159 = Kifogyott a printerből a papír

Ezt a hibát (amennyire tudják) általában kezelik a programok: nem állnak le tőle. Nézzük meg a printert, lehet, hogy csak az adagolás akadt el. Baj lehet a printercsatlakozással is, például szétcsúszhattak a csatlakozók. Ez máskor 160-as hibaüzenetet is okozhat.

162 = Hardverprobléma

Előfordulását a DOS jelzése alapján küldi a program. Hálózatba kapcsolt gépeken jelentkezik a leggyakrabban, ezeknél szinte minden hálózati hozzáférési probléma ilyen hibaüzenettel jár. Ha a program nem kifejezetten hálózati működésre készült, próbáljuk meg futtatását önálló gépen.

200 = Osztás nullával

Ennek az üzenetnek a jelentkezése rendszerint programozási hibára utal, de olyankor is előfordulhat, ha megsérült a program vagy egy .OVR állomány. Az utóbbi esetre azokat a tennivalókat ajánljuk, amelyeknek ismertetésére a 209-es üzenettel kapcsolatban térünk ki.

203 = Halomterület túlcsordulási hibája (Heap overflow error)

Nem túlságosan gyakori hiba, mivel a programozó általában már saját fejlesztőkörnyezetében sem tudja lefordítani a programot, ha ez a jelenség előfordul. Ha mégis találkoznánk vele, próbáljunk meg több szabad memóriát biztosítani a futó programnak a CONFIG.SYS és az AUTOEXEC.BAT fájlok ideiglenes átalakításával.

209 = Overlay-fájl olvasási hibája

Ritkán fordul elő. Az állományok megsérülése okozhatja, .OVR kiterjesztésű fájlok esetében. Győződjünk meg róla, hogy nem sérült-e meg valamelyikük, például másolás közben vagy az installáció során, illetve ha esetleg magunk piszkáltunk bele. Megjegyzendő, hogy az ilyen jellegű hibák gyakran hibaüzenet nélkül, simán a rendszer lefagyásához vezetnek.

216 = Általános védelmi hiba

Hát igen. Nemcsak a Windows képes ilyet produkálni. Védett üzemmódú programoknál (például Turbo Visionban írottaknál) előfordulhat DOS alatt is. Leggyakrabban ilyenkor is programozási hiba áll mögötte, de más okból is jelentkezhet, például védett üzemmódú, grafikus vagy nagy memóriaigényű más programok esetében.

Ilyenkor próbálkozhatunk azzal, hogy olyan környezetben futtatjuk a Borland-féle védett módú bővítést (RTM.EXE + DPMI16BI.OVL) alkalmazó programot, amelyben a DOS-hoz adott EMM386.EXE nincs betöltve a memóriába. Ehhez külön menüpontot is definiálhatunk a CONFIG.SYS és az AUTOEXEC.BAT fájlokban. (Segítségért forduljunk a DOS kézikönyvekhez.)

Azt is érdemes ellenőrizni, hogy az AUTOEXEC.BAT fájlban a definiált keresési útvonalon (PATH=...) nem fordul-e elő több, eltérő verziójú másolata az említett fájloknak.

Simay Endre István

Turbo Pascal programok futásidejű hibakódjai

001 Invalid function number
002 File not found
003 Path not found
004 Too many open files
005 File access denied
006 Invalid file handle
012 Invalid file access code
015 Invalid drive number
016 Cannot remove current directory
017 Cannot rename across drives
018 No more files

100 Disk read error
101 Disk write error
102 File not assigned
103 File not open
104 File not open for input
105 File not open for output
106 Invalid numeric format
150 Disk is write-protected
151 Bad drive request struct length
152 Drive not ready
154 CRC error in data
156 Disk seek error
157 Unknown media type
158 Sector not found
159 Printer out of paper
160 Device write fault
161 Device read fault
162 Hardware failure

200 Division by zero
201 Range check error
202 Stack overflow error
203 Heap overflow error
204 Invalid pointer operation
205 Floating point overflow
206 Floating point underflow
207 Invalid floating point operation
208 Overlay manager not installed
209 Overlay file read error
210 Object not initialized
211 Call to abstract method
212 Stream registration error
213 Collection index out of range
214 Collection overflow error
215 Arithmetic overflow error
216 General Protection fault

Az áprilisi hónap témája az Ifabón

Kin múlik az intranet?

Május 7-én, szerdán délután (a múlt havi számunkban meghirdetett 5 óra helyett) **fél 5 órai kezdettel** kerekasztal vita az Ifabón, a „B" pavilon konferenciatermében. Az Új Alaplap áprilisi hónap témájának nyitott kérdéseiről. A beszélgetés résztvevői:

Halász Gábor (Silicon Graphics)
Klotz Tamás (Oracle)
Molnár Attila (Novell)
Palkó Gábor (Digital Equipment)

Ha az adatbáziskezelőt kezelő kezelő téved

Relációs adatbázisok hibatűrése

Egy adatbáziskezelő annyira biztonságos, amennyire az őt futtató rendszer. A relációs adatbázisok (Oracle, Informix, SQLBase...) állományainak egy része a tényleges adatokat tartalmazza, más része pedig azért felel, hogy konzisztensek legyenek az adatok. Sérülés esetén bizonyos állományok pótolhatók, de vannak olyan fájlok is, amelyeknek elvesztése az egész adatbázis megsemmisüléséhez vezet. A relációs és objektumorientált adatbáziskezelők legfőbb jellemzője (és érdeme) rafinált tranzakciókezelésük és az a tulajdonságuk, hogy meg tudják védeni az adatrendszer integritását a tranzakciók végrehajtása közben.

Hogyan is működik az adat- és tranzakciókezelés? A kliens kezdeményez egy adatmódosító műveletet — legyen ez például az adatbeszúrás („insert"). Azt gondolhatnánk, hogy az adatbáziskezelő rögtön beírja az adatot a megfelelő adatállományba. A szekvenciális adatkezelők (pl. DBF) esetében ez így is van, a relációs adatbáziskezelők mechanizmusa azonban sokkal bonyolultabb. Nekik fel kell készülniük minden előfordulható hardver- és szoftverhibára, és úgy kell kezelniük ezeket, hogy minden körülmények között fennmaradjon az adatok konzisztenciája. Ráadásul egyszerre több lehetséges hibára is számítaniuk kell, és közben gondoskodni arról, hogy ne szakadjon meg a felhasználók kiszolgálása. (Esetleg több száz felhasználóról van szó.) Ha rögtön beíródnának az adatok a helyükre, akkor egy hardver- vagy szoftverhiba bekövetkeztekor törhetnénk a fejünket, hogy melyik pillanatban is fordult elő a hiba. Vajon az adat bekerült-e már akkor az adatbázisba? Vajon megmaradt-e, vagy felborult az adataink integritása? Nos, a tranzakciót éppen ennek a problémának a megoldására találták ki.

A tranzakció tulajdonképpen egy műveletsor, amelyet az adatbázisba bejelentkezett személy (vagy program) kezdeményez az adatbázisban foglalt adatok halmazának megváltoztatására (beszúrására, törlésére vagy felülírására). A tranzakciónak van kezdete és vége. A tranzakció végét a tranzakció lezárásával kezdeményezhetjük, de maga a lezárás is kétféle lehet: igenlő vagy nemleges (commit vagy rollback). Ha igenlő parancs zárja le a tranzakciót, akkor az adatbázistól kapott igen/nem válasz igazolja, hogy valóban módosultak-e az adataink az igényeknek és az előírásoknak megfelelően. Elfogadás esetén mérget vehetünk rá, hogy a kívánt adat a kívánt módon bekerült az adatbázisba (vagy kikerült onnan). Ez így leírva rém egyszerűnek és érthetőnek tűnik, az adatbáziskezelőket gyártó cégek mégis vért izzadnak, hogy megoldják a szükséges műveletsorozatok egyre tökéletesebb, egyre finomabban kidolgozott végrehajtását.

A hibák gyakorisága

Vajon milyen hibákra lehet számítani egy rendszer üzemeltetése során, és hogyan lehet ezeket kivédeni? Három alapvető oka lehet a hibákból eredő adatvesztésnek: a szoftver, a hardver és a felhasználó.

A szoftverhibák jelentik a leggyakoribb hibaforrást. Az adatbáziskezelők rengeteg programsor leírásával készülnek, így elkerülhetetlen, hogy pár ezer soronként néhány tucatnyi hiba bele ne kerüljön még a leggondosabban tesztelt rendszerekbe is. Ehhez még hozzájönnek az operációs rendszerben bujkáló hibák, amelyek ott csücsülnek a memóriában, és alig várják, hogy aktivizálódhassanak. Sajnos a szoftverhibák ellen nincs más védelem, csak a tesztelés. Ennek egy részét a szoftvergyártók hivatásos tesztelőkre bízzák, másik része azonban a felhasználókra vár.

A hardverhibák viszonylag ritkábban okoznak meglepetést, de csak azért, mert a vas öregedésére előre számítani lehet. A hibalehetőségeket prognosztizáljuk, és védekezünk ellenük (tükrözés, RAID, memóriaparitás, fürtözés stb.). A technika eléggé elöl jár a védekezésben, beveti minden elképzelhető fegyverét. Ennek viszont az a hátulütője, hogy ha a technika csődöt mond, akkor csipesszel sem tudjuk összeszedni adatainkat (például a RAID összeomlása esetén).

A hibák elenyészően kis részét okozzák a felhasználók. Vannak persze szórakozott adminisztrátorok, akik véletlenül kiadják a „táblatörlés" parancsot, vagy azért, mert máson jár az eszük, vagy mert nincs kedvük elolvasni a figyelmeztető szöveget. Aki odafigyelés nélkül, csak úgy rutinból püföli az ENTERT, az bármikor könnyen kitöröl egy táblát. Ez is adatvesztés, és ezt csakugyan maga a felhasználó okozza.

Van-e védekezés?

Van-e olyan módszer, amellyel minimálisra lehet csökkenteni a hibák kockázatát? A hibák előfordulását megakadályozni sajnos nem tudjuk, de a veszélyt csökkenteni tudjuk: *mentéssel*. Ha lelkiismeretesen és rendszeresen mentjük adatbázisunkat (hetente, naponta vagy folyamatosan), akkor jöhet szoftverhiba, villám vagy áramszünet, az adatoknak valamilyen korábbi állapotát mindig vissza tudjuk állítani. A később végrehajtott tranzakciók megismétlésével így visszajuthatunk a rendszernek a hiba jelentkezése előtti állapotához.

A mentésnek többféle jól bevált technológiája is van. Az alkalmazott módszerek szinte függetlenek az adatbáziskezelő rendszertől, mindegyikben ugyanazokat a mentési eljárásokat al-

kalmazzák (online, offline, full, incremental).

Nekem személy szerint három adatbáziskezelővel van szorosabb kapcsolatom, ezek közül az Oracle7 áll a legközelebb hozzám, ezért elsősorban ennek a terméknek a „védekező mechanizmusát" és működését említem példaként. (A másik kettő a Centura SQLBase és az Object Design ObjectStore.)

Oracle7 — működés közben

A Oracle cég jól ismert adatbáziskezelője szinte minden platformon megtalálható. Az elmondások alapján kb. tízszer annyi kódot tartalmaz, mint egy átlagos Unix operációs rendszer. Szóval nagy szoftverről van szó, akkor is, ha a működő rendszerből csupán az adatbáziskezelőt nézzük.

Maga a szoftver tulajdonképpen platformfüggetlen. Hogy mégis több platformról elérhető, annak az az egyszerű magyarázata, hogy a cég fejlesztői Sun és a Windows NT környezetben dolgoznak, minden más platformra csak „portolnak". Jóformán minden kód platformfüggetlen módon, C/C++-ban készül, csak azokat a részeket írják más kódolás szerint, amelyek közvetlenül a hardvertől vagy az operációs rendszertől függenek. Ezzel a megoldással sokkal egyszerűbb a szoftverhibák kiszűrése, hiszen egy 90%-ig közös kódot sokkal könnyebb tesztelni, mintha minden platformra külön kódhalmazt készítenének. Ez a magyarázata annak is, hogy az egymásnak megfelelő verziók hibáinak nagy része minden platformon kimutatható. (Leghibamentesebb verzió a híres 7.1.6.2.)

Az 1. ábrán láthatjuk az Oracle7 nagyon leegyszerűsített működési modelljét. Magát az élő adatbázist a program által lefoglalt memória, az adatállomány és a háttérprogramok (processzek) egysége jelenti. Lássuk egy adatbeírás lépéseit:

— Bejelentkeztünk, kiadtunk egy adatmódosítás parancsot (insert, update, delete), indul a tranzakció.

— A szerver beolvassa a szükséges adatblokkokat az adatbázispufferbe.

— A szerver a processzpufferben (memóriában) elvégzi a műveletet, a régi értéket megjegyzi.

— Lezárjuk a tranzakciót (commit). A szerver „elfelejti" a régi értéket.

— Bontjuk a kapcsolatot.

Biztonsági őrizet

Felmerül a kérdés, hogy a memóriában csücsülő adataink kellő biztonsággal kerülnek-e be az adatfájlba. Hiszen bármilyen meghibásodás esetén megtörténhet, hogy az adataink elvesznek.

Az adatkiírás mechanizmusa a következő:

1. lépés. Az LGWR nevű program kiírja az adatokat a naplófájlba. A napló az egyik legfontosabb állomány, szerkezete szekvenciális, elvesztése, sérülése az adatbázis összeomlásához vezethet.

2. lépés. A tranzakció lezárul. Ekkor még csak abban lehetünk biztosak, hogy a naplóállományban meg kell lennie a művelet eredményének, az nem biztos, hogy ki is íródott a adatállományba.

3. lépés. Kiíródik az adatállomány a memóriából a fizikai állományokba (DBWR processz).

Generáljunk mindhárom előző pontnál egy elképzelt hardver/szoftver hibát:

Ad 1. Ha nem tud kiíródni a naplóba a művelet, akkor az a commit (tranzakció jóváhagyása) hibajelzéssel tér vissza. A felhasználó tehát értesül a szomorú tényről, hogy a művelet nem sikerült. Ellenkező esetben a válasz: Igen, a művelet sikeresen végrehajtódott.

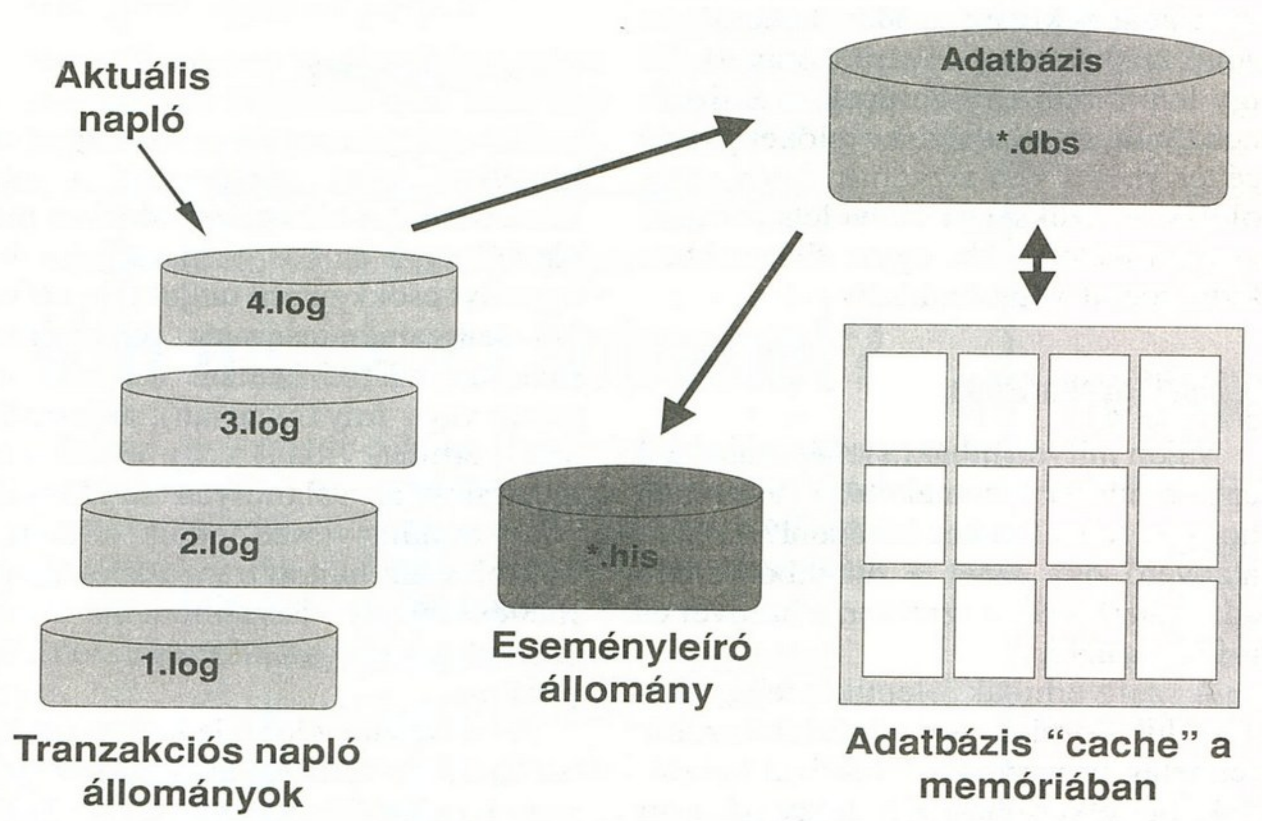
Ad 2. Tranzakció lezárva (sikeresen), de a rendszer közben elszáll. Mi történik ilyenkor, hiszen a rendszerrel együtt az adat is elszáll?! A szerver a rendszer újraindításakor automatikusan beolvassa a tranzakciószámot (a rendszer által generált azonosítót), és konstatálja, hogy nem konzisztens az adatbázis. Elővesz a naplóállományt, és elvégzi a benne bejegyzett műveleteket. Vagyis újra lejátssza az adatmódosítást. Ez igen gyorsan történik (pár másodpercet vesz igénybe), hiszen csak az értelmezett műveleteket kell sorra beírogatni.

Ad 3. Adatkiírás közben is történhet hiba — a szerver ekkor is a 2. pontnál leírt módszert alkalmazza. Miután az adatkiírás megtörtént, a szerver (DBWR) beállítja az aktuális tranzakciószámot, ezzel jelezve, hogy a művelet fizikailag is szerencsésen végetért. Leegyszerűsítve valahogy így játszódik le a fizikai adatvédelem.

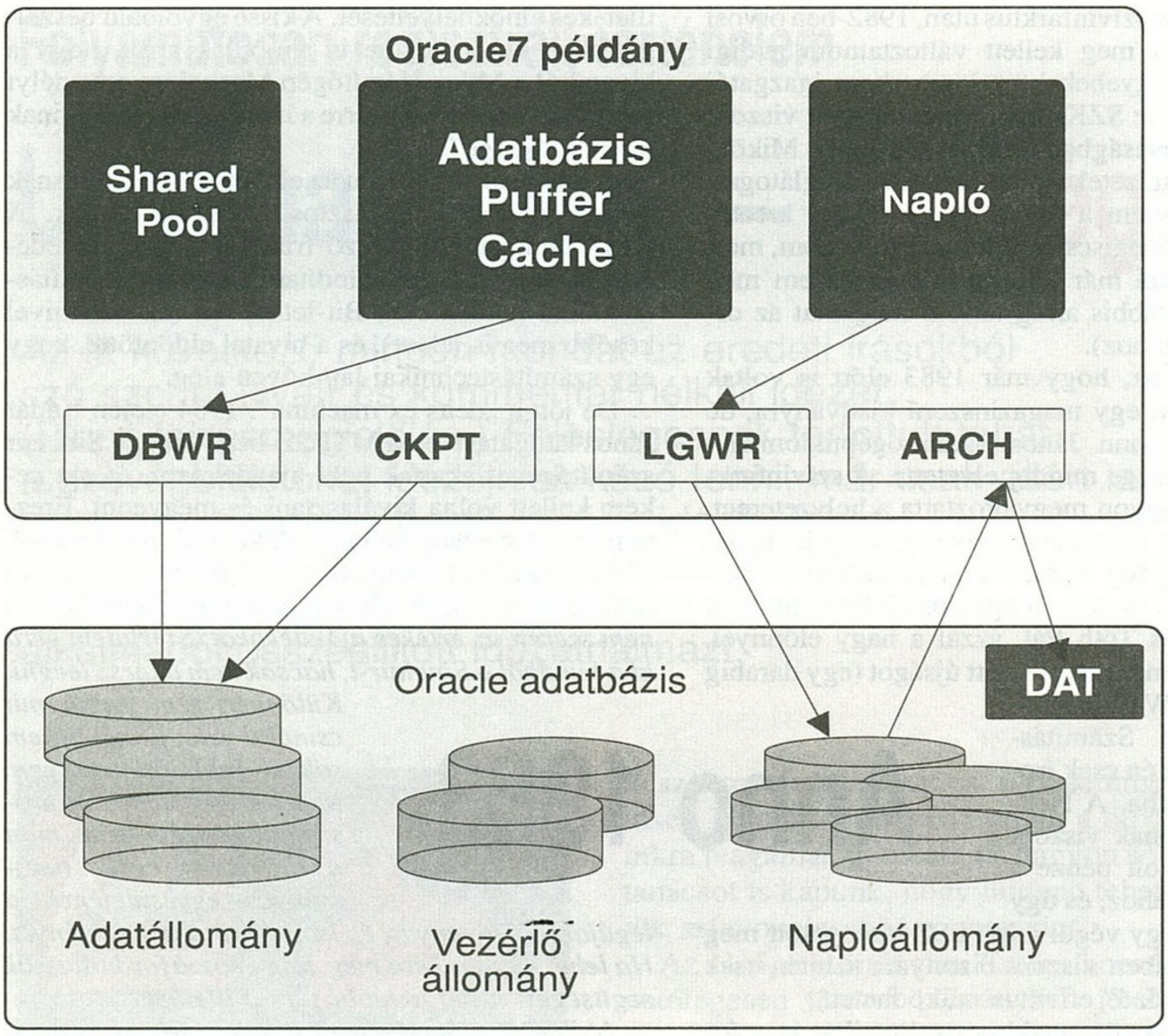
A pótkötél pótkötelének pótkötele

Mi történik, ha elvész vagy megsérül a napló vagy az adatállomány? Ilyesmire bármikor számíthat a felhasználó, ellene tehát különböző biztonsági intézkedéseket kellett bevezetni. Az Oracle7 támogatja az állományok tükrözését, és kötelezően elő is írja, hogy a legfontosabb állományokat tükrözni kell (napló!). De ha valaki még a tükrözésben sem bízik, az a naplóállományokat is folyamatosan mentheti a rendszer segítségével (ARCH processz). Ez automatikus, csak biztosítani kell számára a megfelelő méretű helyet.

A rendszeres mentésekkel kiküszöbölhetjük (vagy legalábbis minimálisra csökkenthetjük) az adatvesztést. A mentések automatizálhatók, nem igényelnek állandó felügyeletet. Csak arra vigyázzunk, hogy úgy ne járjunk, mint az egyszeri adatbázis-adminisztrátor, aki felülírta a jó mentést a már hibás állománnyal...



1. ábra



2. ábra

Szólni szeretnék még olyan fontos rendszerekről, amelyeknél súlyos összegekbe kerül minden másodperc (bank, posta, légiközlekedés stb.). Ilyen helyzetekben a szerverek tükrözése a legjobb védekezés a hibák ellen. Két számítógép működhet párhuzamosan. Ezek a folyamatos üzemmódot úgy biztosíthatják, hogy a két gép bármikor helyet tud cserélni. Mivel e rendszereket leállítani sem szabad, a mentéseknek is mindig üzem közben (online) kell történniük.

Akik szeretik rendszeresen figyelemmel kísérni rendszerük (oprendszer + adatbázis) összhangját, vagy optimalizálni szeretnék a működését, azok legjobban teszik, ha speciális szoftvert szereznek be erre a célra. Igen jól megfelel például a Platinum Enterprise Performance Monitor, a Tsreorg, a Fast Unload.

Ezek segítségével nemcsak monitorozhatják és optimalizálhatják rendszerüket, hanem egyes hibák forrását is könnyebben felfedezhetik.

SQLBase

Az SQLBase relációs adatbáziskezelő rendszer fiatalnak tekinthető a piacon, de zsenge kora ellenére elég jól megállja a helyét. Inkább kisebb adatbázisok kezelésére tervezték (1-2 GB). Elsődleges futtató platformja a Windows NT, a Novell Netware vagy a Unix-világból a Sun Solaris. Teljesítménye és tudása semmiben sem marad el a nagy óriások mögött. Minket azonban most inkább csak a felépítése és adatkezelési koncepciója érdekel.

Lássunk róla egy egyszerűsített ábrát. A 2. ábrán megfigyelhetjük, hogy működési elve hasonlít az Oracle7-nél látottakhoz. (Ez nem a véletlen műve. Ugyanaz a Gupta úr alkotta meg az SQLBase-t, aki az Oracle 6. verziójának alapjait lerakta. Ő a Centura szoftvercég alapító tulajdonosa.)

Az SQLBase-ben is megtalálhatjuk a naplóállományt, az eseményleíró fájlt, az adatműveletek itt is a memóriában történnek. A hasonlóság ellenére mintha a megvalósítás egyszerűbb lenne, anélkül, hogy ez csökkentené a hatékonyságot.

A legérzékenyebb pontja ennek a rendszernek is a napló elvesztése vagy megsérülése. Erre nagyon oda kell figyelni, mert nélküle nincs élet. Az adatbázisnak ez annyira szerves része, hogy ugyanúgy vigyázni kell rá, mint az adatokra. Legjobb megoldás itt is a napló tükrözése (vagy külön merevlemezre való elhelyezése), és természetesen a rendszeres mentés.

ObjectStore

Az ObjectStore ugyanúgy tranzakciókezeléssel működik, mint minden vállalati adatbáziskezelő rendszer, függetlenül attól, hogy relációs elven alapul-e vagy objektumorientált felépítésű. Ha összeomlik az adatbáziskezelőt működtető szerver (vagy a gép, vagy az operációs rendszer), akkor újraindítás után az ObjectStore automatikusan biztosítja, hogy újrarendeződjön az adatbázis, és felálljon az utolsó rekonstruálható konzisztens állapot. Ugyanez érvényes az adatbáziskezelő klienseire is. A nem jóváhagyott (committált) tranzakciók utolsó maradványai is kisöprődnek az adatbázisból.

Az ObjectStore esetén az adatbázisok között közvetlen kapcsolatokat (pointer, referencia) lehet létesíteni. Így egy logikai (elosztott) adatbázis tetszőleges számú adatbázisfájlból állhat. Az adatbázis mérete is tetszőleges lehet. 1-2 GB méretig közvetlen pointerek alkalmazhatók, ez pedig igen gyors működést tesz lehetővé. Ennél nagyobb méretű adatbázisok esetén referenciákon keresztül oldható meg az összekapcsolás.

A programozó számára a hibakezelés a C++ „exception"-jeinek formájában jelenik meg. Az ObjectStore „exception" osztályai bőséges információt adnak a hiba környezetének meghatározására. Megoldható az is, hogy saját hibakezelésünket hozzáigazítjuk és integráljuk az ObjectStore „exception" mechanizmusához. Arra viszont vigyáznunk kell, hogy egy olyan kliens program, amelynek inkompatibilis képe van a megnyitandó adatbázisról, az ObjectStore természetéből fakadóan megnyitni sem tudja azt. Az adatbázisgazda feladata, hogy (viszonylag egyszerű eszközökkel) kifejtse az adatbázis sémát. ObjectStore adatbázis esetén — a relációs SQL adatbázisoktól eltérően — egy kliens program soha nem szállhat el olyan hibák miatt, amelyek az adatbázis és a kliens program verziójának inkompatibilitásából fakadnak (például „table or view does not exist").

Az ObjectStore igen terjedelmes eszközkészlettel rendelkezik. Saját eszközei vannak az adatbáziskezelő működésének monitorozásához és diagnosztizálásához, akár API, akár önállóan működtethető utility programok formájában. Külön termékként beszerezhető a Performance Expand is, amely kifejezetten az ObjectStore adatbázisgazdáinak a számára készült.

Firnága László
Firnaga@iqsoft.hu

15. évfolyam, 150. szám

Az Új Alaplap 1997-ben lépett 15. évfolyamába, és kalandos életútján a mostani szám a 150. állomás. Egy hagyományos tematikájú lap esetében ezek az „adatok" semmilyen rekordot nem döntögetnének, de a számítástechnika maga is nagyon fiatal, publikációs fórumai között sehol sincsenek matuzsálemek: az egyik legrégebbi és talán mindmáig a legtekintélyesebb, legszínvonalasabb folyóirat, a Byte is csak fennállásának 20. évfordulóját ünnepelte a múlt évben. Lapunk tehát, amely Mikroszámítógép Magazin néven 1983-ban indult, nemzetközi viszonylatban is az úttörők közé tartozik. Laptársai, a külföldi lapok első hazai változatai csak több évvel később kerültek fel a magyar sajtópalettára.

A 150. szám apropóján a fontos állomásokat jelképező címlapokból és a laptörténeti tényeket rögzítő korabeli írások részleteiből összeállítottunk egy kis montázst, így a „később beszállt" olvasók is áttekintést kaphatnak erről a kalandos útról. A lapindulás és a Mikroszámítógép Magazin korszak néhány kulisszatitkába Kovács Győző írása ad további bepillantást. Az ő lelkesedése és kitartása nélkül folyóiratunk aligha született volna meg, a Mikroszámítógép Magazin időszakot mindvégig felelős szerkesztőként jegyző Könyves Tóth Pállal közösen kialakított koncepciójuk és szerkesztési gyakorlatuk pedig döntően meghatározta a lap első 7 évének jellegét, s abból az Alaplap szerkesztőségi csapata a második 7 évben sok értékeset megőrzött.

Az első mikroszámítógépes generáció azért is kötődött annyira a „Mikromagazinhoz", mert a szakmai tudásanyagon, a „biteken és bájtokon" kívül értékes humán tartalmat is kapott tőle. Az [Új] Alaplap a rohamos technikai változásokat követve és a gazdasági élet követelményeihez igazodva tartalmilag is, szerkezetileg is többször átalakult, de gondolkodásmódban, szellemi kalandvágyban a hagyományokat folytatta — és folytatni is akarja.

Anno 1983

Egy súlyos szívinfarktus után, 1982-ben orvosi tanácsra meg kellett változtatnom addigi életvitelemet. Egyebek közt lemondtam igazgatói beosztásomról az SZKI-ban. Megtartottam viszont a Neumann Társaságban főtitkári tisztemet. Miközben a gyógyintézetek sorát kellett „végiglátogatnom", azon törtem a fejem, hogy valami kreatív tevékenységet mégiscsak jó lenne folytatnom, mert bár a szívemmel már baj van, de az eszem még működik (legalábbis amíg tartom magamat az orvosi instrukciókhoz).

Az igazság az, hogy már 1983 előtt is voltak előterjesztéseim egy magazinszerű kiadványra, de azokat a Neumann János Számítógéptudományi Társaság elnöksége mindig elvetette. A szívinfarktus azonban nagyon megváltoztatta a helyzetemet, mert én lettem a „szegény Győző, aki bizonyára hamarosan meg fog halni, hadd teljesülhessen utolsó kívánsága". A terv megvalósításában mellém társult Könyves Tóth Pál, azzal a nagy előnnyel, hogy előzőleg már szerkesztett újságot (egy darabig a mostani CW-Számítástechnika előtti Számítástechnikát), míg én csak írogattam a lapokba. A Neumann Társaságnak viszont nem nagyon volt pénze a lap megindításához, és úgy emlékszem, hogy végül a MTESZ támogatott meg bennünket, amiben viszont bizonyára szintén csak a „szegény Győző" effektus működhetett.

Olvasmányos, ugyanakkor szakmailag is színvonalas számítástechnikai magazint akartunk csinálni, elsősorban a fiataloknak, a diákoknak, hiszen akkor nem volt olyan lap, amelyből számítástechnikai ismereteiket gyarapíthatták volna. Könyves Tóth Pál lett a felelős szerkesztő, aki a lap tartalmát és megjelenési arculatát határozta meg, én pedig a szerkesztőbizottság vezetője, aki részben összefogta a szerkesztőséget, részben pedig intézte a lap kapcsolatait, és persze írt is abba.

Megszerveztük az első szerkesztőséget, és kiválasztottuk a rovatvezetőket. Talán álljon itt az első számot szerkesztő és író nagyon lelkes közreműködő gárda neve: Broczkó Péter (hírek), Budai György (személyi számítógépek), Garádi János (feladatok, rövid és ravasz programok), Jakab Ágnes (ember és gép kapcsolat), Kovács Győző (vezércikk és levelezés), Nacsa Sándor (termékismertető), Pataki Ernő (programozástechnika), Petróczi Judit (könyvek), Pogány Csaba (alkalmazástechnika, tanfolyam), Simonyi Endre (klub), Szabó János (reklám), Varga András (iskola-számítógép), Vass Nándor (alkalmazások) és Votisky Zsuzsa (játékprogramok).

Első és legfontosabb feladatom az volt, hogy megszerezzem a lapengedélyt. Miközben maga a lapkészítés már sínen volt, a kormány Tájékoztatási Hivatalának ügyintézőitől olyan információkat kaptunk, hogy két számot „kézirat gyanánt" megjelentethetünk ugyan, a továbbiakhoz szükséges lapengedélyt azonban nem fogjuk megkapni.

Az első szám így lapengedély nélkül, 1983 őszén jelent meg, mint az NJSZT „kiadványa", ugyanis nem utalhattunk annak folyóirat jellegére. Volt ugyan benne egy egyedi ISBN szám (amit persze a másodikon már nem lehetett azonosítóként feltüntetni), az ISSN-re (a periodikák jelölésére) viszont semmi kilátásunk nem volt.

Kerestem a „szocialista összeköttetéseket", de csak rosszabbnál rosszabb tanácsokat kaptam. Végül felhívtam telefonon a nagytekintélyű hivatal illetékes elnökhelyettesét. A kissé egyoldalú beszélgetésben kifejtette: ne is álmodjak arról, hogy a hivataltól a Mikroszámítógép Magazinra engedélyt fogunk kapni, ugyanis erre a lapra „a társadalomnak nincs szüksége".

A jólértesültek szerint az elutasító magatartásnak sokkal kevésbé „magasztos" indítékai voltak. A KISZ támogatását élvező friss (és persze engedélyezett) lap, az Ötlet is indítani akart egy számítástechnikai mellékletet, Bit-let néven (az valamivel később meg is jelent), és a hivatal eldöntötte, hogy egy számítástechnikai lap bőven elég.

De jött a „deus ex machina". 1984 elején Kádár János látogatást tett a MTESZ-ben. Ott egy Sinclair számítógépet akartak neki ajándékozni, és azt nekem kellett volna kiválasztani és megvenni. Éreztem, hogy itt a nagy pillanat. A MTESZ főtitkárának némi malíciával a következőket mondtam: *„Ugye tudod, hogy Kádár elvtárs igen puritán ember, és nem szereti az értékes ajándékokat. Szerintem verd ki a fejedből a Sinclair-t, hacsak nem akarsz leégni. Különben sem tudna mit csinálni vele. Ennél nekem sokkal jobb ötletem van. Mutasd meg neki a Mikroszámítógép Magazint, mint a szövetség egyik tudományos egyesületének a legújabb kiadványát, és kérd ki róla a véleményét. Ha lehet, arra is kérd meg, hogy hozzá fordulhassak segítségért a lap további sorsát illetően."*

Akik a beszélgetésen ott voltak, másnap elmondták, hogy a lapnak óriási sikere volt a pártfunkcionáriusok körében, különösen, amikor látták, hogy Kádár János nemcsak élvezettel forgatja a lapot, hanem dicséri is. És Kádár mondott egy nevet (XX), akit *„a főszerkesztő elvtárs, ha kell felhívhat, és ő biztosan mindenben segíteni fog."*

Másnap késedelem nélkül felhívtam — no nem XX elvtársat, hanem a Tájékoztatási Hivatal elnökhelyettesét, és megkérdeztem tőle, mikor kapjuk meg a lapengedélyt. Éreztem, hogy erre a szemtelenségre a másik oldal „lilára vörösödik". A telefonkagylót legalább húsz centire kellett tartanom a fülemtől, de a „soha" szóban összegezhető választ annyira nyugodtan hallgattam végig, hogy ezen kissé meghökkent. „Érti, amit mondok?" — kérdezte. „Értem" — mondtam —, de tegyen meg egy szívességet, hívja fel XX elvtársat, és vele is közölje ezt."

Csönd a másik oldalon, ezért most már én kérdeztem meg: „Itt van még?" Mire ő: „XX elvtársat a Kádár titkárságon?" Én: „Igen, ott. Csak azért kérem, hogy hívja fel, mert nekem biztosan nem hiszi el, hogy tegnap Kádár elvtárs, akinek odaadtuk a lap első számát, megdicsérte a Mikro-t, és azt mondta, hogy ez kell a magyar ifjúságnak." (A történelmi hűség kedvéért hozzáteszem, hogy Kádár ez utóbbi megjegyzést nem mondta ugyan, de végül is mondhatta volna.) Nem telt bele fél óra, és most már az elnökhelyettes telefonált: „No jó, jöjjenek be a lapengedélyért." Mire én: „Nem tudná kiküldeni?"

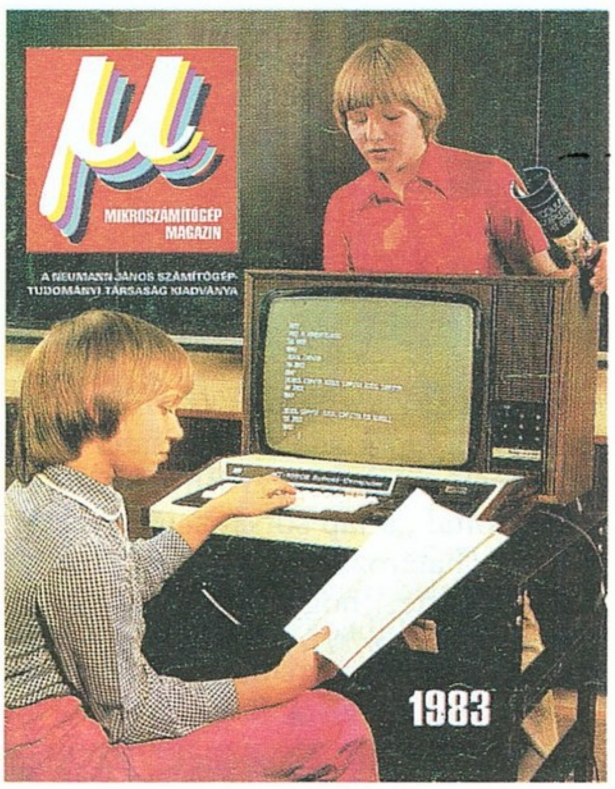
Kiküldte — és a Mikroszámítógép Magazin ezzel jogilag is megszületett. A folytatás ettől kezdve hasonlít a fordulatokban bővelkedő megannyi többi laptörténetre. Aki ezekben részt vett, annak természetesen minden sokkal izgalmasabb volt. Ha azonban érdekli a Mikromagazin korszak, akkor „lapozza fel" visszaemlékezésemnek a lemezmellékleten folytatódó további „fejezeteit" is.

Kovács Győző

Folyamatosan regisztrált történelem

Ugyanaz a lap — és mégsem ugyanaz

Az itt következő minden mondat az eredeti írásokból szó szerint kivett és kommentár nélküli idézet. (Plusz forrásmegjelölés.) Feleslegesnek tartottuk tehát a szövegblokkokat idézőjelek közé tenni, már csak azért is, hogy ne zavarják a belső, stiláris idézőjeleket. A táblázatok adatai és a képaláírások pedig tényszerűen kiegészítik a lapunk 150. számának megjelenése alkalmából összeállított idézethalmazt.



A nevezetes legelső szám

Indul a μ Magazin, társaságunk új kiadványa. ... Azt szeretnénk, ha a diák és a szakember, a hobbiból programot író vagy otthon számítóget építő amatőr, a tanár és a mérnök, egyszóval mindenki, akit a számítástechnikai problémák és ezen belül is elsősorban a személyi számítógépek érdekelnek, megtalálná benne a számára érdekes, a munkáját vagy szórakozását segítő számítástechnikai megoldásokat, tanácsokat. *(Kovács Győző, Mikroszámítógép Magazin, 1983/1.)*

✱

Egy olyan fórumot akartunk teremteni, amelynek célja a számítástechnikai kultúra tömeges terjesztése, tehát szakmai igénnyel elsősorban a nem számítástechnikai szakemberekhez, az amatőrökhöz és a diákokhoz kívánt szólni. ... Az olvasói levelekből kiderült, hogy nemcsak az amatőrök, hanem a szakma is olvassa és figyelemmel kíséri lapunkat, és nem csupán a hibáinkra reagálnak gyorsan, de nagyon sok tanácsot is kapunk, hogy hogyan tehetjük színvonalasabbá a magazint.

Az első, a „történelmi" szám hozzávetőlegesen tízezer példányban jelent meg. Ma az olvasói kör már stabilizálódott és a lap általában húszezer körüli példányban kerül az olvasókhoz. Nem titok, hogy amióta megkezdődött a gazdasági „szabályozók" drasztikus változása, valamint az áremelések sorozata a nyomdaiparban, azóta állandó gondunk a lap gazdasági stabilitásának a fenntartása. *(Kovács Győző, Mikroszámítógép Magazin, 1988/12.)*

✱

Rengeteg pótgazdánk volt az állandóan és megszorítás nélkül támogatást nyújtó Neumann Társaság mint főgazda mellett. Egy darabig támogatott bennünket a Művelődési Minisztérium, a boldogult emlékű TII kezével adta azt, amit adott, volt egy rövid kalandunk a KISZ KB-val, de végül is hosszan, magas hőfokú szerelemmel senkinek sem kellettünk. Váltogattuk a kiadókat is, végigpróbáltuk a Deltát, az Ifjúsági Kiadót, volt egy szerencsétlen kalandunk a Reformmal is. *(Kovács Győző, Mikroszámítógép Magazin, 1990/4.)*



A művészi címlapok korszaka



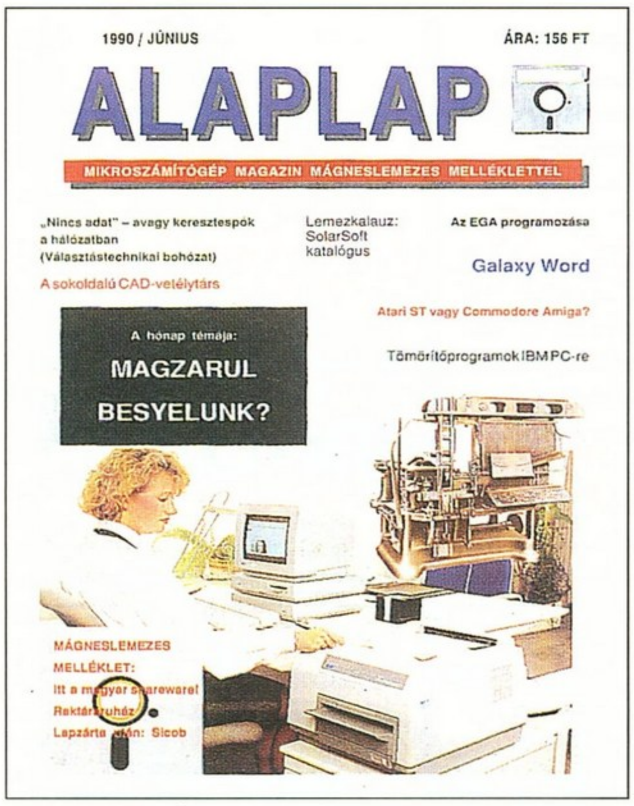
A megújulás előtti utolsó Mikro

✱

A fordulat akkor következett be, amikor 1990 elején egy dinamikus, új szemléletű kisvállalat, a Cédrus Informatikai Rt rászánta magát, hogy megkísérli a lap megmentését, átveszi a szerkesztőséget, viseli az átmeneti veszteségeket, megpróbálja gazdaságossá tenni a lapot. Erre természetesen csak teljesen új koncepció alapján lehetett esélye, amit a Neumann Társaság vezetősége először el is fogadott, s jelképes mértékű anyagi terhet maga is vállalt a megújhodásban. Az előkészületek során azonban a lapgazda egyre inkább úgy érezte, hogy a születendő gyermek nem igazán az övé, s mielőtt az átalakítás első terméke, a júniusi szám megjelent volna, az NJSZT már meg is tagadta apaságát. *(Faklen Pál, Magyar Sajtó, 1990/12)*

✱

A kiadó kollektívája és a szerkesztőség úgy látta, hogy a megoldás egyetlen lehetséges útja a „menekülés előre".



Átváltozás Alaplappá

Valami mást, többet, egyedülállót kell nyújtani az olvasóknak — a lehető legalacsonyabb áron.

Mit is módosítottunk a Mikroszámítógép Magazinon?

Alaplap. Új név választása már önmagában is vállalkozás. A javaslatok közül végül is a Mikromagazin és az Alaplap volt versenyben, s az utóbbi mellett kötöttünk ki — remélhetőleg helyesen építve annak játékos kétértelműségére, magyarosságára és számítástechnikai jelentőségére.

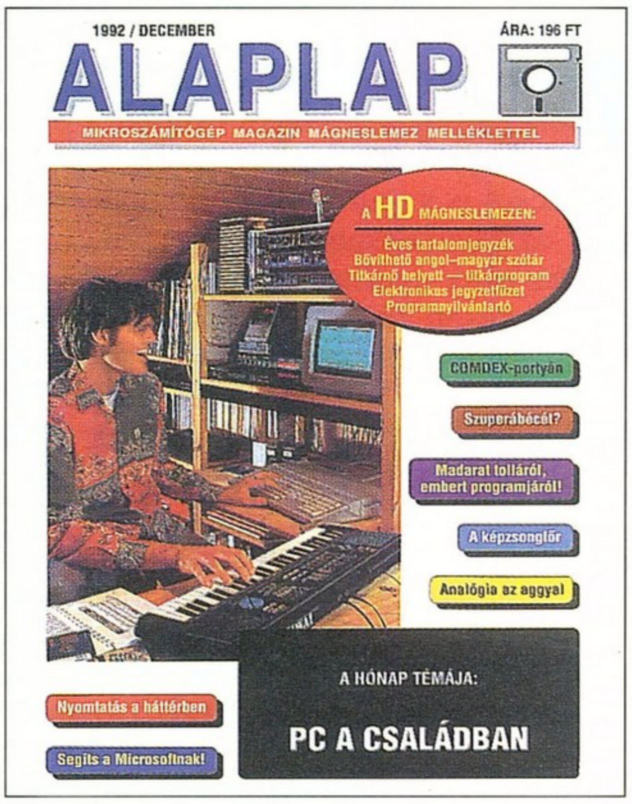
Mágneslemezes melléklet. Az Alaplap elnevezés nem egészen új, mert már egy éve megjelenik ezzel a névvel a Cédrus Kiadó egyik mágneslemezújságja, s azt váltotta most fel állandó mellékletünk. Ezzel a szolgáltatással egyedülállóak vagyunk a hazai számítástechnikai sajtóban.

SolarSoft. A közprogramok, vagyis az olcsón hozzáférhető szoftver (shareware, freeware, public domain) ismertetésével foglalkozott a SolarSoft Magazin, amely most mint önálló kiadvány szintén megszűnik és lapunkba beolvadva állandó rovatainkat gazdagítja.

Kikhez szól az Alaplap? Szándékunk szerint elsősorban a személyi számítógépek használóihoz, akik nem számítástechnikusok, de akiknek munkájához hozzátartozik a számítógépek intelligens alkalmazása; akiknek nem hobbijuk a programozás, de akik ismerni szeretnék a technikai eszközök és programok lehetőségeit és korlátait. De nyilvánvalóan szól az Alaplap a számítástechnikusokhoz és a számítástechnikai eszközök gyártóihoz és forgalmazóihoz is, hiszen nekik legalább olyan fontos, hogy sokirányú visszacsatolást kapjanak a felhasználói igényekről, problémákról, megoldásokról, a számítástechnika itthoni alkalmazásának hétköznapi valóságáról. *(Faklen Pál – Tölgyes Péter, Alaplap, 1990/6.)*

✷

A Cédrus Kiadó Kft és az IDG között mostani számunk lapzártája után megkötött szerződés értelmében az Alaplap 1993-tól az IDG kiadványaként jelenik meg. ... Az új laptulajdonos nem kíván lényegesen változtatni az Alaplap jellegén és stílusán, így munkatársi gárdánkat is teljes egészében átvette *(Alaplap, 1992/12)*



Az első HD-lemezes szám

✷

Szakmai körökben felbukkant a hír, hogy az IDG csupán a lap „lenyelésére", az olvasótábor, az újságírók, a technikai eszközök megszerzésére készül, de a lap kiadását hosszabb távon nem finanszírozza. Ezt Bíró István azzal cáfolta: „a lap sorsát saját nyereségessége határozza meg. Nem azért fizettünk érte, hogy megszüntessük". *(Dombi Gábor, Népszabadság, 1992. december 7.)*

✷

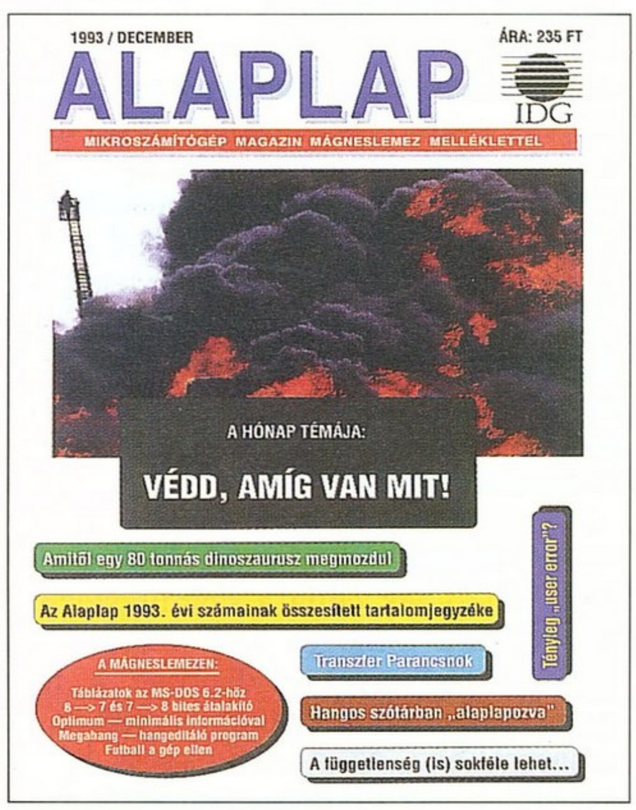
A Cédrus Rt „csődjének" az Alaplapra vetett árnyéka, a háttérben folyó adásvételi tárgyalások, a napi sajtóban

A lap példányonkénti ára:

Időszak/lap	Ár
1983/1	18,50
1984/1 – 1984/6	28,–
1985/1 – 1989/12	30,–
1990/1 – 1990/5	31,–
1990/6 – 1991/1	156,–
1991/2 – 1992/12	196,–
1993/1 – 1993/12	235,–
1994/1 – 1994/12	279,–
1995/1 – 1995/5	297,–
1995/6 – 1995/11	356,–
1995/12	456,–
1996/1 – 1996/3	356,–
1996/4	495,–
1996/5 – 1996/9	356,–
1996/10	496,–
1996/11 – 1996/12	396,–
1997/1	496,–
1997/2 – 1997/4	396,–
1997/5	586,–

Mellékletek:

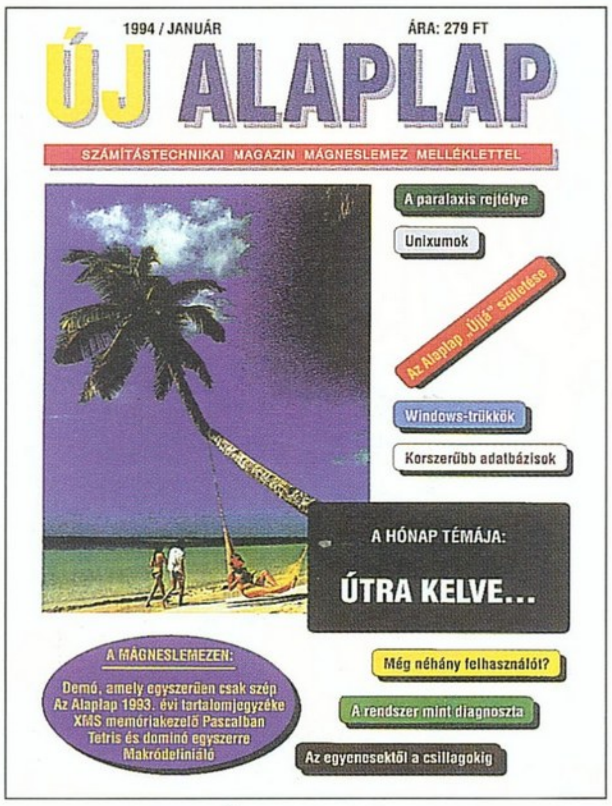
5,25" DD floppy: 1990/9 – 1995/2
(2 db: 1991/11, HD: 1992/12)
5,25" HD floppy: 1995/3 – 1995/12
3,5" HD floppy: 1996/1 –
Extra CD-ROM: 1995/12, 1996/4, 1996/10, 1997/1, 1997/5
PC-Papír: 1990/12, 1991/12, 1995/2



Az IDG-korszak záróakkordja — üzenettel

Lapnév	Időszak	Lapgazda/Kiadó	Felelős szerkesztő
Mikroszámítógép Magazin	1983/1—1990/5	NJSZT***	Könyves Tóth Pál
Alaplap	1990/6—1992/12	Cédrus	Faklen Pál
Alaplap	1993/1—1993/12	IDG	Faklen Pál
Új Alaplap	1994/1—	Új Alaplap	Faklen Pál

*** A Neumann János Számítógéptudományi Társaság mellett közel két évig (1986-88) a lap társtulajdonosa volt a KISZ KB is. Az NJSZT a kiadót többször változtatta, ezek időrendben: MTESZ, Lapkiadó Vállat, Delta Szaklapkiadó, Ifjúsági Lapkiadó, NJSZT, Cédrus. Utána viszont már a laptulajdonos és a kiadó mindig egybeesett.



„Újjá" születve

pontatlanul megjelenő hírek, és nem kis részben a szerkesztőség meg a kiadó kétszeri hirtelen elköltözése miatt egy ideig zavar és „kommunikációs hézag" vett körül bennünket, jövőnket illetően pedig számos találgatás kapott lábra.

A mindenkit érdeklő alapkérdés az volt, hogy megmarad-e az Alaplap. Eleinte mi is bizalmatlanok voltunk, hátha a konkurencia felvásárlásának kemény amerikai módszerét alkalmazzák velünk szemben, s az IDG rejtett szándéka az Alaplap megszüntetése vagy a PC World-be való beolvasztása, hogy ezáltal megszabaduljon legjobb, legeredetibb, legnagyobb példányszámú konkurensétől. Ennek a feltevésnek azonban volt egy alapvető gyengéje: a PC World és az Alaplap nem átfedi, hanem inkább kiegészíti egymást. ... Sokkal valószínűbbnek tartjuk tehát, hogy az IDG palettájának bővítésére törekszik, az Alaplapot pedig — annak egész koncepciójával együtt — meg akarja tartani. *(Faklen Pál, Alaplap, 1993/1)*

✱

Szűkebb körben egy hónapja az sem titok már, hogy 1994 januárjától az Alaplap a PC World szerves részeként folytatja pályafutását. ... Nem szűnik meg az IDG által tavaly megvásárolt Alaplap: legjobb rovataiban fog tovább élni. *(Mester Sándor, Computerworld-Számítástechnika, 1993. november 9.)*

✱

A régi „alaplapos" gárda nem fogadta el az IDG-nél, az Alaplapot magába olvasztani szándékozó PC World-nél számukra fölkínált állást, hanem ... megalakította saját kiadóját, s januártól — mintha mi sem történt volna — kiadják a régivel minden lényeges tulajdonságában megegyező, ám jogi okokból újnak nevezett Új Alaplapot! *(VGA Monitor, 1993. december)*

✱

Az Alaplap olvasótáborának a laphoz való erős kötődése, gyakran megnyilvánuló rokonszenve és támogatási készsége a legfőbb támpontunk az „Újjá" születéshez. ... Az Alaplaphoz szokott olvasókat egyébként aligha vezeti félre az „Új" megnevezés, hiszen már az első szám után tudni fogják, hogy ez a lap „igazándiból a régi". Ilyen lapnévvel mégsem mertünk előállni, nehogy az újságárusoknak dilemmát okozzunk, amikor a „Régi Alaplapot" kérik tőlük. *(Faklen Pál, Új Alaplap, 1994/1)*

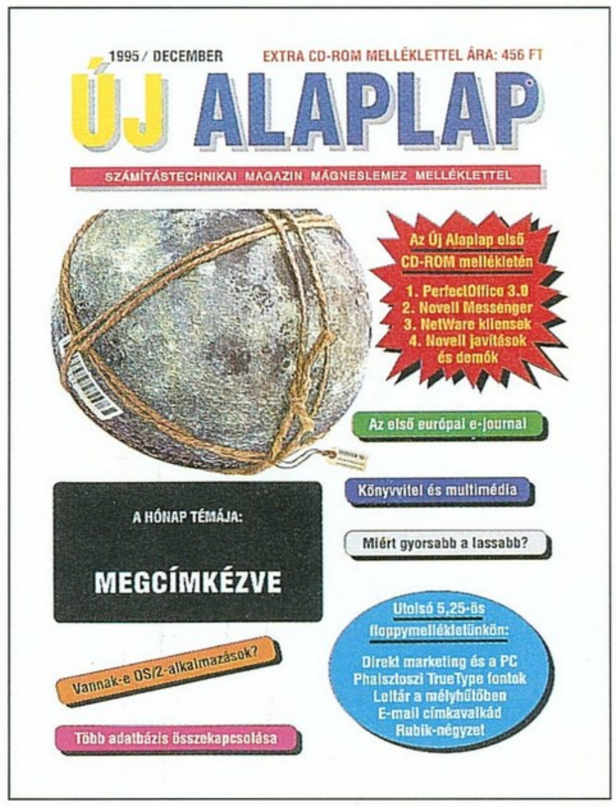
✱

Az Alaplap „törzsgárdatag" hirdetői szerencsére azonnal „vették a lapot", hogy az Új Alaplappal egy már bejáratott, jól bevált média áll továbbra is rendelkezésükre, ezért nemcsak bíztak terveink megvalósíthatóságában, hanem hirdetésükkel már rögtön az első számban mellénk álltak.

A végére maradt az egyik legfontosabb visszacsatolási folyamat, amelyet rendszeres és alkalmi cikkszerzőink, illetve programíróink alakítottak ki velünk, és a jelek szerint a kiadóváltás után is töretlenül fenntartanak. Szakmai hitvallásunkhoz tartozik, hogy minden értékes szellemi alkotásnak biztosítsuk a nyilvánosság előtti megmérettetés lehetőségét. Erre ugyanúgy szüksége van egy programozónak, mint egy költőnek. És leginkább szüksége van rá az egész társadalomnak! Az Új Alaplap ehhez ad fórumot, akárcsak 1983-tól 8 éven át a Mikroszámítógép Magazin, majd az utóbbi 3 és fél évben az Alaplap tette, és ha a költőket nem is, a számítástechnikában eredeti, alkotó módon gondolkodókat arra biztatjuk, hogy továbbra



Humorral jobban megy...



Az első szám extra CD-ROM-mal

is juttassák el hozzánk közlésre érdemesnek tartott munkáikat. Nem ígérjük, hogy mindent megjelentetünk, mert szerencsére bőven van miből válogatnunk, de a legjobbak előbb-utóbb eddig is sorra kerültek. Szabad a pálya. *(Faklen Pál, Új Alaplap, 1994/2)*

✱

Viszonylag magas, 28,8 százalék azoknak az aránya, akik már a Mikroszámítógép Magazin időszaktól (1983–1990) hűséges olvasóink.

Kérdőívünk egyik pontja azt igyekezett kideríteni, hogy számítástechnikai felkészültség, tudásszint vagy jellegzetes tevékenységi kör szempontjából most milyen összetételű lapunk olvasótábora. Az „öndiagnózis" alapján kialakult megoszlás:

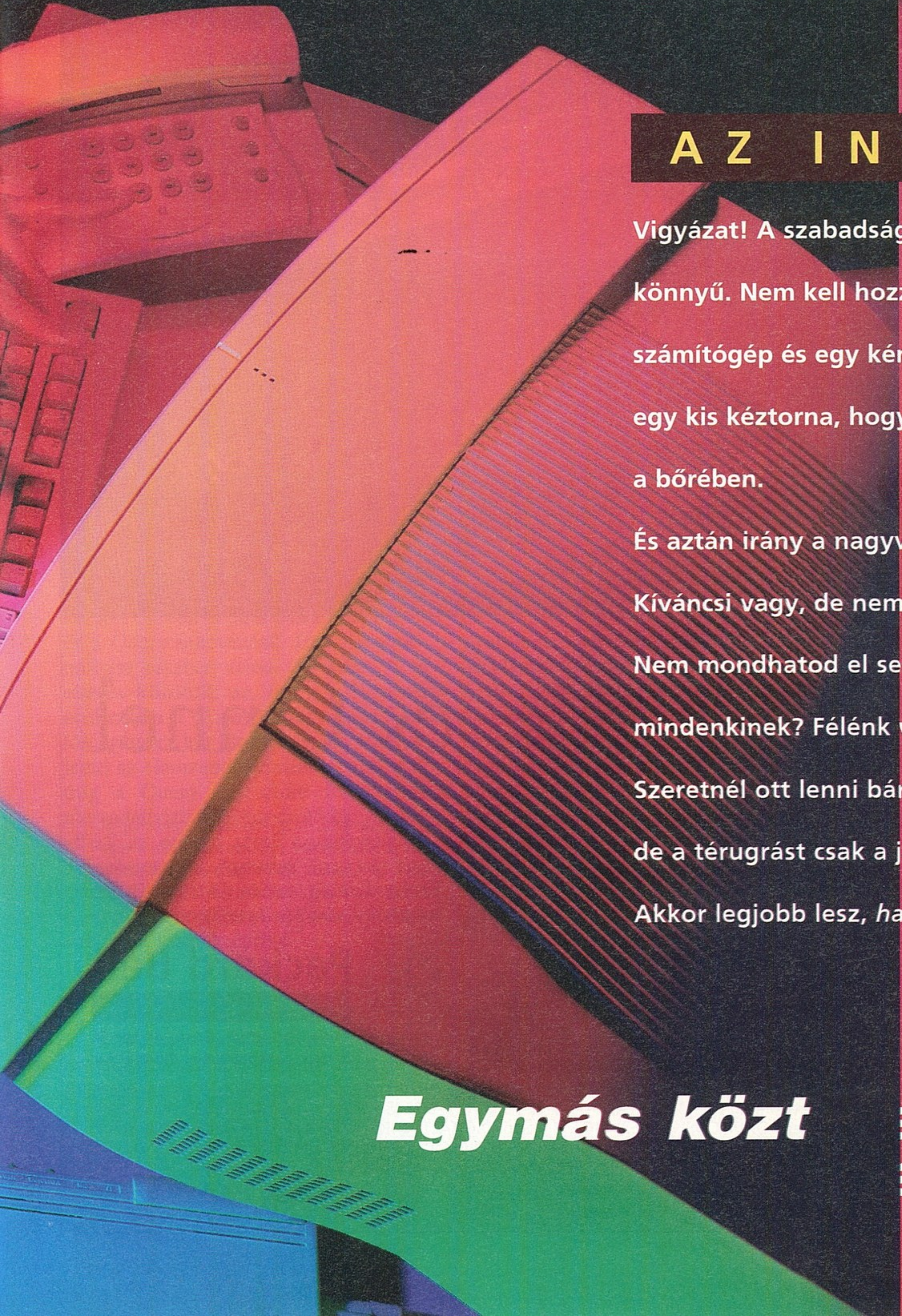
Kezdő felhasználó	12,6%
Haladó felhasználó	22,4%
Tapasztalt felhasználó	16,5%
Számítástechnikát tanuló	12,1%
Szoftverfejlesztő, programozó	13,8%
Hardverfejlesztő és -szakértő	2,8%
Rendszergazda, -szervező	9,4%
Számítástechnikai kereskedő	1,8%
Számtástechnikai oktató, kutató	7,4%
Egyéb	1,2%

A válaszok megerősítették azt a személyes tapasztalatunkat, hogy az Új Alaplap nem egyetlen vagy néhány homogén réteg olvasmánya. Kezdő felhasználóktól a profi számítástechnikusokig tulajdonképpen minden réteghez szól. Persze azon belül nem mindenkihez, hanem minden rétegnek a leginkább tudásvágyó, az új gondolatok iránt nyitott, a számítástechnikát más szakterületekkel összefüggésben szemlélő, szakmai véleményirányító részéhez. *(Faklen Pál, Új Alaplap, 1996/4)*


MATÁVnet
Ingyenes információs zöld szám:
06 80 203 203
Már 2000,- Ft havi előfizetési díjért elérhető
a MATÁVnet-en a korlátlan szabadság.

HALLÓ itt a korlátlan szabadság

AZ INTERNETEN



Vigyázat! A szabadság rabjává válni végtelenül könnyű. Nem kell hozzá más, csak egy telefon, egy számítógép és egy kényelmes fotel. Reggelente egy kis kéztorna, hogy az egér is jól érezze magát a bőrében.

És aztán irány a nagyvilág. Irány a MATÁV**net**.

Kíváncsi vagy, de nem akarsz hamar megöregedni? Nem mondhatod el senkinek, de elmondanád mindenkinek? Félénk vagy, de hódítani akarsz? Szeretnél ott lenni bárhol, ahol csak kedved tartja, de a térugrást csak a játszótérről ismered? Akkor legjobb lesz, *ha nyitsz egy pár ablakot.* Klikk.

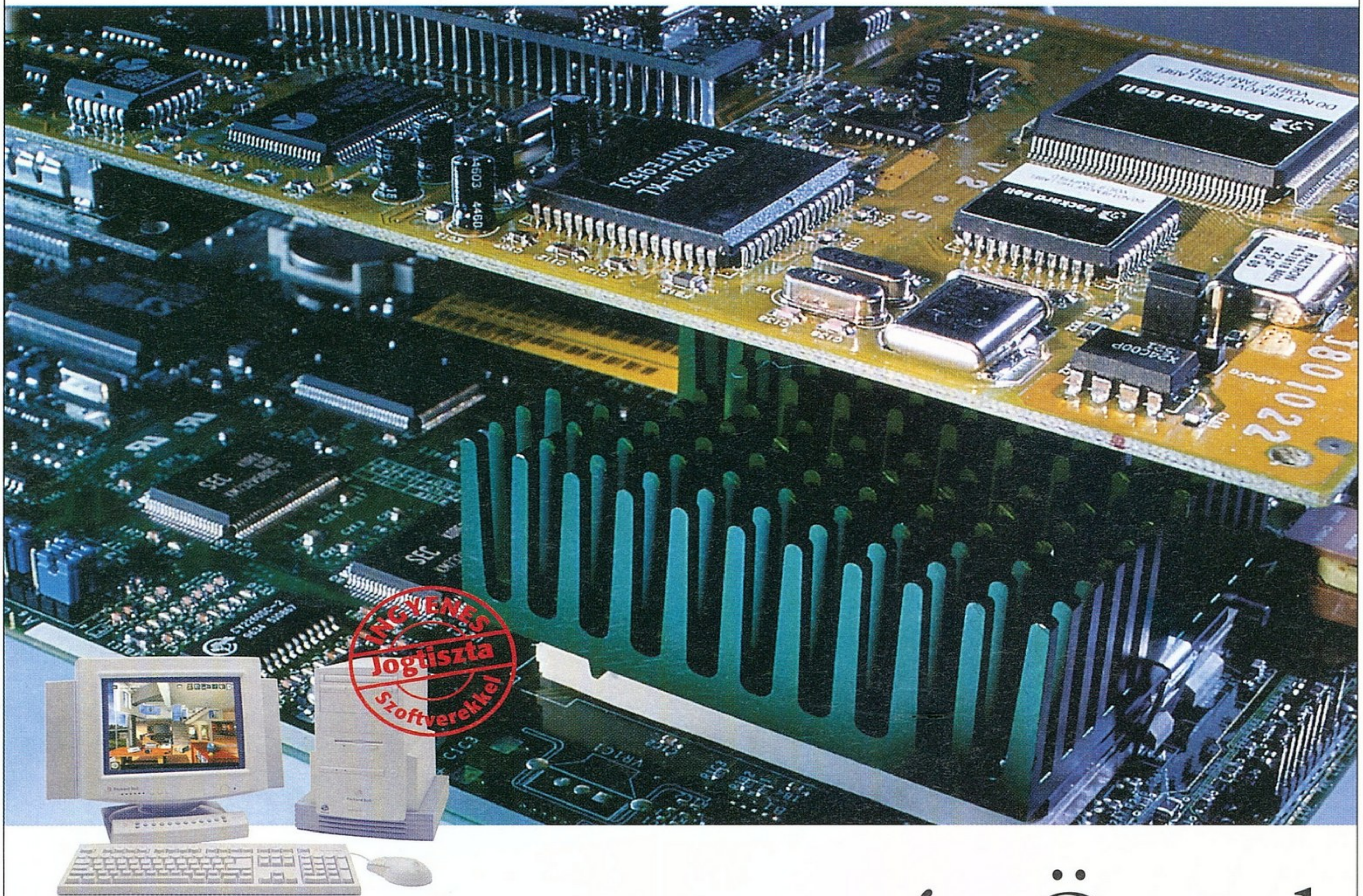
Egymás közt

MATÁV

LOWE GGK

INFORMÁCIÓKÉRÉS: 0529

Valamennyi alkatrészünk tökéletesen együttműködik egymással ...



Pentium 120 MHz, 16 MB RAM,
1.27 GB HDD, 8x CD, 3D támogatás,
16 bites hangkártya, rádiókártya, faxmodem,
aktív hangszóró, alacsony sugárzású monitor,
Windows '95 (magyar) operációs rendszer,
+ 6-os CD-csomag

... és Önnel

Az otthoni számítógépek eladásában a Packard Bell – a világ egyik legnagyobb számítógépgyártója – az első.
Sikerünk titka az, hogy tudjuk, a számítógépeinknek kell megérteniük és követniük a felhasználók igényeit. Gépeink vannak az emberekért és nem fordítva.
Számítógépeink rendkívül sok szolgáltatást nyújtanak: **Internet,** CD, fax, telefon, tv, rádió, videó stb. A komputerek egyre szélesedő felhasználási területe, illetve a szoftverek növekvő memóriaigénye azonban folyamatos bővítést jelentenek. Mi nemcsak a felhasználást könnyítjük meg – a számítógépeinkhez adott ingyenes, jogtiszta szoftverekkel –, hanem arra is odafigyelünk, hogy igényeinek állandó növekedését is egyszerűen tudja követni. Amikor Ön a Packard Bell mellett dönt, akkor biztos lehet abban, hogy a későbbiek során is gond nélkül tudja számítógépét használni. Fejlesztéseink során különösen ügyeltünk arra, hogy minden egyes új alkatrész tökéletesen együttműködjön a régiekkel és Önnel. Ezért számítógépeinkben később sem fog csalódni. A Packard Bell megbízható társa lesz hosszú éveken át.

Packard Bell
Otthon a számítástechnikában
1149 Budapest, Bosnyák tér 5.
E-mail: packbell@mail.datanet.hu

Budapest MÉDIA MARKT (1) 347-1650 • interPC networking Kft. (1) 251-9678 • OFFICE PLUS (1) 270-2545 • KERAVILL (1) 465-1002 • PC Soft Kft. (1) 120-3442 • UNICOMP Kft. (1) 217-4170 • **Debrecen** FŐNIXCOMP Kft. (52) 410-571 • **Győr** DIGITÁLTECHNIKA (96) 414-411 • **Hódmezővásárhely** DELFIN Computer Rt. (62) 346-033 • **Kaposvár** MICROKAL Kft. (82) 320-513 • **Kecskemét** HC Fair Kft. (76) 482-186 • **Miskolc** dr. LAN (46) 412-368 • NET Rendszerház Kft. (46) 411-412 • **Nagykanizsa** HC Byte Kft. (93) 314-333 • **Pécs** COMPUTER-TECHNIKA Kft. (72) 450-207 • HC Pointer Kft. (72) 310-662 • **Székesfehérvár** UNICOMP Kft. (22) 327-446 • **Szekszárd** KVANTUM Kft. (74) 419-541 • **Tatabánya** ALFADAT (34) 310-234

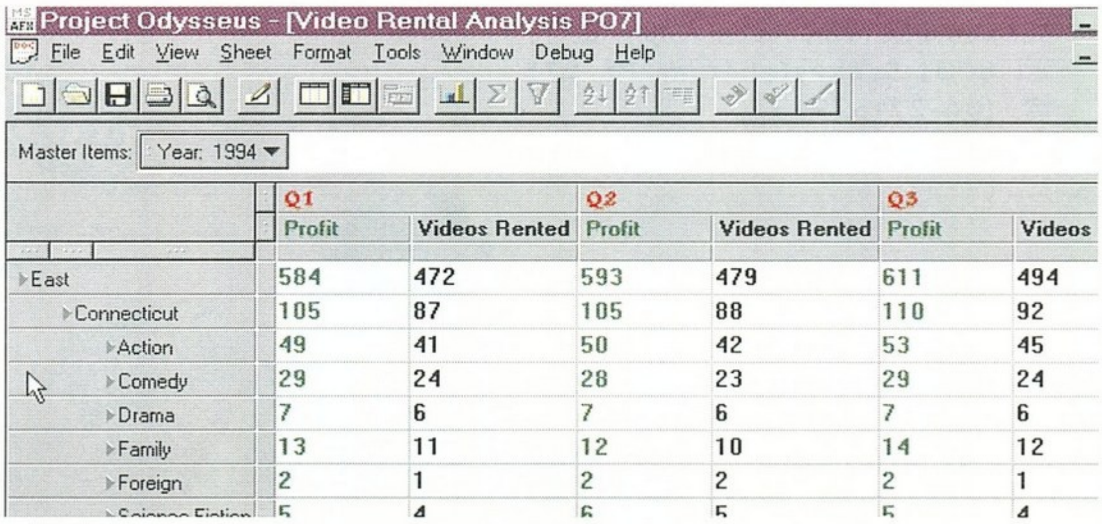
Hot line: (1) 252-0545

STRAM-AVANT Budapest

INFORMÁCIÓKÉRÉS: 0537 ▲

Discoverer/2000

Adatanalízis



A relációs adatanalízis eszköze az Oracle Discoverer/2000 termékcsalád, amelynek legújabb verziója a 3.0-s. A termék segítségével a felhasználók komoly számítástechnikai ismeretek nélkül is képesek olyan lekérdezéseket összeállítani, amelyekkel nagyon mély elemzőmunkát lehet végezni.

Az Oracle Discoverer/2000 szoftver fő feladata, hogy segítségével a relációs adatbázisban lévő adatokat oly módon lehessen prezentálni a felhasználóknak, hogy maga az eszköz elrejtse a relációs adatbáziskezelőben lévő elemzőséma komplexitását, és megfelelően egyszerű kezelői felületet biztosítson. Mindez a ma egyre inkább feljövőben lévő analízis-iparág korszerű ismérveinek való megfelelést jelenti: az adatba való lemerülés, az adatkivonás, az adatkifejtés, az adatbányászat, az adatnézet-forgatás eredményeit anélkül kell szolgáltatni, hogy ennek érdekében a felhasználónak bonyolult SQL utasításokat kellene összeállítania.

Az Oracle Discoverer/2000 termékcsalád előző verziójában a lekérdezéseket két eszköz is támogatta: a Browser és a Query elnevezésű. Ezek funkcionalitása részben átfedi egymást. A fő eltérés közöttük az, hogy a Browser elsősorban a gyakorlottabb felhasználók számára lett kialakítva, így közvetlenül az elemzendő adatsémában kutat, míg a Query egy előre felépítendő metaadatbázishoz kapcsolódik, amely elfedi az eredeti séma komplexitását. Természetesen namcsak az ad hoc queryt támogatják az eszközök, hanem a jelentéskészítő (report) funkcionalitások is be vannak építve.

Operatív, relációsan

A mintegy másfél éves Odysseus projekt fejlesztési eredményeinek megfelelően az új, 3.0-s verzió mint analíziseszköz-frissítés azt célozza, hogy az Oracle relációs adatbázisára is egyre inkább alapuljanak olyan eszközök, amelyek lehetővé teszik nagy mennyiségű operatív működési adat relációs alapú elemzését. Elsősorban azoknak a felhasználóknak fontos ez a termék, akik nem szándékoznak rögtön egy adatáruház-koncepció szerinti átfogó adatanalízis-megoldást kiépíteni.

Olyan felhasználókról van szó, akik számára elegendő megoldásnak tűnhet az operatív rendszer jelentéskészítő lehetőségeinek kiegészítése egy olyan eszközzel, amely hatékonyan támogatja az ad hoc intuitív lekérdezéseket. Ez a megoldás sok esetben elegendő lehet kisebb cégeknél vagy nagyobb cégek leányvállalatainál, részlegeinél, és ezt szem előtt tartva, a 3.0-s termékváltozat egyik újdonsága az, hogy amennyiben egy teljesen gyakorlatlan felhasználó leül a képernyő elé, és elkezd dolgozni, akkor munkáját speciális „varázslók" (wizardok) segítik. Megmutatják, hogyan kell definiálni például, ha egy táblázatból kíván adatokat kinyerni, vagy azt, ha az oszlopok megjelenítési formáját kívánja megváltoztatni. Az Oracle a fejlesztéskor — még mielőtt egyetlen sornyi forráskód is elkészült volna — igénybe vette a tesztelő partnerek segítségét, és az ő javaslataikat figyelembe véve alakította ki a felhasználói interfészt.

Metaadatrétegek

Ugyanakkor itt, az új verzióban nemcsak a végfelhasználói interfészről van szó, hanem a korábbi verziókban kissé elhanyagolt, kevesebb segítségben részesült adminisztrátorok is kapnak wizard-segítséget, azaz ezeken a varázslókon keresztül definiálhatják azt a metaadatréteget, amelyen azok a felhasználók dolgozhatnak, akiknek nem kell látniuk a metaadatréteg által elfedett adatkomplexitást.

Gyakorta előfordul, hogy amennyiben relációs adatbáziskezelővel szemben futtatunk egy komplex lekérdezést, a lekérdezés végrehajtása elhúzódik. A Oracle Discoverer/2000 3.0 viszont lehetőséget nyújt arra, hogy befolyásoljuk az erőforrás-felhasználást. Még ennél is fontosabb, hogy a lekérdezés végrehajtása előtt a Discoverer/2000 3.0 mintegy 40 százalékos pontossággal képes megbecsülni a lekérdezés időtartamát, ami után további indexek bevonásával valamelyik struktúrát úgy alakíthatjuk át, hogy csökkentsük a lekérdezés idejét. Az új verzió nagy valószínűséggel megmondja, hogy adott esetben miért lassú egy lekérdezés, mindent auditál, információt gyűjt minden lekérdezésről. Amennyiben egy kritériumrendszer segítségével úgy dönt, hogy elő kell állítani és tárolni kell egy összesített (aggregált) adathalmazt, akkor javasolja, hogy az adminisztrátor hozzon létre egy új táblát, amelyben az a nagyszerű, hogy mindez automatikusan történhet. Az adminisztrátor javaslatot kap a Discoverer/2000 3.0-tól, hogy összegtáblát kellene előállítani. Ha az adminisztrátor jóváhagyja azt, akkor a Discoverer/2000 megbecsüli a szükséges helyet, létrehozza az összegtáblát, és gondoskodik annak frissítéséről, majd az adminisztrátor megszabja, hogy a Discoverer/2000 a továbbiakban milyen időközönként végezze a táblán az aggregálást.

A Discoverer/2000 új verziója által támogatott adatlefúrást és adatkibontást kiegészíti egy ún. hyperdrill, egy hiperkibontó képesség, amely lehetővé teszi, hogy a végadatcellára rámutatva, rákattintva különböző adattípusokat — akár videót, akár hangot, akár Wordben vagy Excelben készült dokumentumokat — hozzunk föl. S még egy fontos tulajdonság: az Oracle általános intranet-koncepciójához illeszkedve ez a verzió alkalmas arra is, hogy analízislapokat, elkészült elemző táblákat tegyünk fel a Webre, az Internetre és az intranetekre.

Csányi György

Kritikai vitriológia

A kukac karrierje

Tavaszi hírcsokor vadvirágokkal és vadhajtásokkal, valamint talajmenti fagyok helyett lefagyásokkal...

Úgy tűnik, a Microsoft a PC-s piacon mind hegemónia, mind pedig dominancia szempontjából felülmúlta az IBM-et. Node mi várható a mainframe-piacon? A Microsoft immár itt is meg akarja vetni a lábát (vagy inkább bekopog, és a résre nyitott ajtóba udvariatlanul beerőszakolja a lábát). Cedar, azaz magyarul Cédrus kódnév alatt készül az a szoftver, amely arra hivatott, hogy a Windows NT-s és Windows 95-ös gépek mint kliensek integrálhatóak legyenek egy nagygépes rendszerbe. A lecke úgy hangzik, hogy a Microsoft az eleddig csak NT szerveren futó Microsoft TransAction szervert (MTA, de semmi köze a Magyar Tudományos Akadémiához) közösíti az IBM rivális tranzakció-processzorával, a CICS-szel (Customer Information Control System), amely pedig nagygépeken, illetve AS/400-on, RS/6000-en és OS/2 alatt fut. A CICS NT-s változatát addig is szorgalmasan fejlesztgeti az IBM.

A Cedar valószínűleg nem marad sokáig magányos cédrus, mert az év végére esedékes szoftverkapcsot valószínűleg további, ezúttal alkalmazás szagú szoftverek is követni fogják. Bár mint F. R. David kb. 10 éves slágeréből tudjuk: Words (and Excels) don't come easy.

Hol a Linux?

Olvasom, hogy 1996 volt az az első év, amikor a világszerte telepített Windows NT-s gépek száma elérte és meghaladta a Unix operációs rendszert futtató gépekét. Fájdalmas tény ez bizony a HP, a DEC, a Sun, az SCO és a többi unixos elkötelezett számára. De vajon beleszámolták-e a Linuxokat is a statisztikába?

Megjelent az első spéci asztali gép, amely már 4,7 Gbájtos DVD-meghajtót tartalmaz. Ez pedig a Toshiba cég Infinia 7220-as modellje, amely ráadásul 200 MHz-es MMX technológiával is felvértezett Pentiumot hord a szíve helyén, egyúttal rádió- és tévé-tuner is jár hozzá, és csupán 4000 dollárba kerül (potom 800 000 jó magyar forint). A Toshiba nem várta meg, hogy a Microsoft mérnökei végre kiizzadják és közreadják a DVD 32 bites drivereit, megírták ők maguk.

FAT-halál az Új Technológiával

Az alcím kifejtése: adataink elvesztése NTFS partícióra formázott harddiszkünkről, mert a Windows NT makacsul nem kíván a továbbiakban bootolni. Sokakkal előfordult már. Eddig két út kínálkozott a rendszergazdák számára, úgymint ad 1: az NT újratelepítése; ekkor adott esetben keresztet is vethettünk adatainkra. Ad 2: lásd ad 1, majd egy előző archív mentés visszatöltése. Utóbbi esetben csak az utolsó mentés után keletkezett adatok hiánya okozhat fejfájást.

Ha nem NT-vel lenne dolgunk (sem a Windows 95 OSR2 FAT-32-jével), mindössze annyi lenne a teendőnk, hogy a gépet egy DOS-os bootlemezzel újraindítanánk, az adatokat kényelmesen elmentenénk, majd új rendszert telepítenénk stb... De a fránya NT NTFS-partícióját csak maga az NT olvassa. Az pedig nem fér el egy floppyn. Róka fogta csuka. Segít egy ügyes shareware program, amelynek a neve: NTFSDOS. Megtalálható és letölthető például a következő helyről az Interneten: www.ntinternals.com/ntfsdos.htm.

Tehát a teendő a fenti esetben ezek után imígyen fog kinézni: tetszés szerinti DOS-ról bootolni egy floppydiszkkel és elindítani az NTFSDOS-t. A backup szoftverek sajnos körmönfontabbak annál, mintsem hogy egy ilyen kis program megzabolázza őket. Azok direkt módon próbálják meg kezelni a harddiszkvezérlőt, megkerülve az operációs rendszer diszkkezelésre vonatkozó szolgáltatásait, s teszik mindezt a nagyobb sebesség érdekében. Így tehát nem közvetlenül tudunk backupolni, hanem mondjuk egy párhuzamos kábel segítségével a read only állapotban lévő NTFS partícióról átvisszük az adatokat egy másikra gépre, méghozzá a DOS-os LapLink vagy a Norton Commander Link funkciója segítségével. Ennyi.

Vizuális monstrum

Winfax Pro 8.0, a várvavárt, Windows NT-n is működő, valódi, 32 bites faxprogram. Amely végre egynél több modemet is tud kezelni egyidejűleg. (Persze ha egynél több telefonvonalunk lenne a faxolásra...) Hálózatban is működik, beáll a sorba a Winfax Pro Server 4.1 klienseként, de már kapható a Winfax Pro 8.0 Server változat is. Nehéz szülés volt ez a Symantecnél. Több mint két évig várt rá a világ.

És megint Microsoft. (Pedig higgyék el, nem fizet azért, hogy az újság ennyiszer kiírja a nevét.) Tehát a lényeg: a Microsoft megjelentette minden idők legösszetettebb, mondhatni a számítástechnika szinte minden területét felölelő fejlesztőeszköz-monstrumát, a Visual Studio 97-et. A legutolsó hírek szerint 5 CD-n pöffeszkedik, de idők múlásával ez a szám csak emelkedhet. Kész arzenál. Benne foglaltatnak a Microsoft legfőbb fejlesztőnyelvei, úgymint: Visual C++ 5.0, Visual J++ 1.1 és Visual Basic 5.0, továbbá a legújabb Internettekergető, a Visual InterDev 1.0, adatbáziskezelőnek a Visual FoxPro 5.0, forráskódkövető és karbantartó, ún. verziókontroll rendszernek a Visual SourceSafe, végül kliens/szerver típusú adatbázismotornak — hoppá, ez nem Visual!!! — az SQL Server 6.5 (ami Microsoft köntösbe bújtatott SyBase, mint tudjuk). Mielőtt bárki aggódni kezdene, megnyugtatom, hogy az Office 97 Professional sem hiányzik ebből az óriáscsecsemőből, és az apróságokat már meg sem említem. (Talán egyszerűbb lett volna azokat a termékeket felsorolni, amelyek nem fértek bele a Visual Studióba.)

Mindazonáltal a Microsoft célja, azt hiszem, egyértelmű: nemcsak a fejlesztőeszközök minél nagyobb fokú integrálása, hanem az egyes individuális programozók, rendszertervezők egyetlen nyájba gyűjtése és kihajtása a

Microsoft-alapú mezőre. Ahol — valljuk be — egyelőre dúsan terem a fű, bár tele van pitypanggal. Majd gazol a nép, Gates pedig erdőirtással újabb legelőterületekhez jut.

Harcban a Corellel a túlélésért

Másfél évvel ezelőtt írtam a Micrografx cég ABC Graphics Suite nevű integrált termékéről, hogy ez a cég utolsó dobása: mindent egy lapra feltéve gyakorlatilag összes műveit rápréseltette egy CD-ROM-ra, és olcsóbban adta, mintha bármelyik komponensét egymagában vennénk meg. Nos, a terv, úgy látszik, bevált. A termék jól fogyott, márciusban kijött a következő Suite, ezúttal azonban elhagyták a névből az ABC megjelölést, helyette a cég nevének állítanak emléket: Micrografx Graphics Suite2, igen, így egybeírva a Suite és a kettes. A tartalomjegyzék: Micrografx Flowcharter 7.0, Picture Publisher 7.0, Designer 7.0, Simply 3D 2.0, valamint egy teljesen új MediaManager, meg az elkészített image-ek közvetlen Internet-kapcsolódását, azaz weblapra emelhetőségét biztosító QuickSilver technológia.

Egy jópofa hely a Weben: Why Windows 95 Sucks!, ízes magyarsággal: Miért szivat a Windows 95! Kanada leghíresebb egyetemén, a Waterloo Universityn található ez a hely: www .csclub.uwaterloo.ca/~vtluu/ww95s.

Ide azok látogassanak el, akiknek már tele a hócipőjük a Windows 95-tel. Vicces írásokat, karikatúrákat fog találni a kíváncsi Win95-gyűlölő. És persze alternatívákat (a szivatásra...). Érdekes módon konkrétan a Windows NT-t ajánlja kivezető útnak.

Összerakni, szétvagdosni, össze...

Megjelent a Travelling Software műhelyéből (tudjuk, ők a LapLink atyamesterei) a WebEx 2.0, amely egy off-line Internet-böngésző. Az új változat már Explorer-szerű kinézettel bír, parancssor, ablakok stb. Ami érdekes még, hogy upgrade a korábbi termékről NINCS. Kategorikusan nincs. Ezzel a cégmentalitással is meg kell barátkoznia az embernek. Igaz persze, hogy a WebEx kb. 30-40 dolláros termék. Az egészen kicsi cégek éltek eddig ezzel a terjesztési móddal. Egy vagy két saját fejlesztésű termékkel rendelkeznek, korábbi vásárlóikat vagy kedvezményben részesítik, vagy nem. A vagy nem jó ürügye, hogy ugyanazt a terméket kissé továbbfejlesztve másképpen nevezik el: például az Alien Skin cég Photoshop plug-injének korábbi neve Black Box 2.0 volt, de ezután nem a 3.0 következett, hanem az Eye Candy. Más, hasonló módszer a fejlesztőeszközöket gyártó cégek esetén a különböző platformokat támogató eszközök hol össze-, hol pedig szétcsomagolása, ezzel is kiszűrhetők az upgrade-elgető, spórolós ügyfelek. Lásd a kanadai Sequiter Inc. CodeBase nevű terméke, ahol pillanatnyilag különváltan él a C, a C++, a Basic és a Delphi/Pascal nyelvhez készített adatbáziskezelő könyvtárgyűjtemény.

Újdonság a Watcom C/C++ 11.0 is. Az egyetlen forráskód-független, komplex fejlesztőcsomag, amely 16 és 32 bites Windowsokhoz, valamint Novell szerveren futó alkalmazáshoz, illetve OS/2 alá egyaránt tud kódot készíteni. A legtöbb böhönc játékprogram Watcom C-vel készül. A hasonlóképpen nem apróság AutoCAD is.

Marketing és versenyhivatal

Az America Online (AOL) mint Internet-szolgáltató egyszerűen túlnőtte magát. Túl sikeresnek bizonyultak a kedvezményes előfizetési akciók (20 dolcsiért havonta korlátlan hozzáférési lehetőséggel). Egy éve még szinte minden újságban találhattunk AOL próbaszoftvert és x órányi ingyenes hozzáférési lehetőséget. A felhasználók boomja azt eredményezte, hogy az addig viszonylag gyors elérésű hálózat kis híján összeomlott (Clinton beavatásának online közvetítése közben ez meg is történt, miként itthon az Internetto Horn-interjúja idején is...), a reakcióidők percnyire duzzadtak. A hot-line leterheltsége is félórás holtidőket eredményezett a segélykérők türelmét próbálandó. Az amerikai versenyhivatal kötelezte is azonnal az AOL-t, hogy addig nem regisztrálhat új előfizetőt, amíg megfelelő hardverberuházással fel nem készül a hirtelen több mint 8 milliósra nőtt előfizetői tábor kiszolgálására. Addig is sokan azzal segítenek korábbi kedvenc szolgáltatójuknak — ami innen nézve persze pikáns asszociációkat ébreszt —, hogy átmenetileg átjelentkeznek máshoz. A piaci egyensúly így ismét helyreáll. Ne ijedjünk meg, az ott nem Magyarország.

Marketing, oh. Az Intel a Pentium Pro processzora utáni családot eleddig a nem valami jó csengésű Klamath névvel illette. Mivel közeleg a proci hivatalos megjelenése is — amely igazából az MMX technológiával kiegészített Pentium Prónak fog megfelelni, s a nyár folyamán 233 MHz vagy gyorsabb órajellel ketyegve jön ki —, az Intel főreklámosai úgy döntöttek, hogy maradnak a már több mint 3 éve szajkózott — azaz bocsánat, jól bevezetett — név mellett (amellyel kábeltévés csatornák reklámjaiban is gyakorta találkozhatunk), így lett az új/régi név Pentium II.

Házi keresőgüzü

Végül hadd mutassak be egy olyan vívmányt, amely a Netről származott vissza a lokális, nem feltétlenül Weben lógó PC-re. A trendek irányt váltanak? Ami valahol jó és működik, miért ne lehetne érte pénzt is kérni? Az AltaVista keresőgépei által összegereblyézett információhalmaz sok online Internet-keresőszoftver alapja. Ilyen például a Yahoo is. Szavak, szófoszlányok alapján keresgélhettünk a végeláthatatlan Web-rengetegben. Azután eszébe jutott valakinek, hogy az AltaVista nagypályás változata mellett életképes lehet annak kispályás, helyi gépi erőforrásokon kutató változata is. Hát a névválasztás (hosszú fájlnevek kezelése ide vagy oda) itt sem tűnik túlságosan telitalálatnak: Search My Computer Private eXtension 1.0 (így, nagy X-szel a szó belsejében). Kb. 30 dollárt kérnek érte.

A módszere ismert: először végiggázol mindenen, amiben szöveges információ található. A szöveget indexeli (egy 1 GB-os wincsi 3-5 percig tarthat), majd a feltett kérdésekre gyakorlatilag kapásból válaszol. Az Inso Quick Viewjával is összepasszintható, mellesleg természetesen a Netscape Navigator és a Microsoft Internet Explorer mellett is szimbiózisba léphet más alkalmazásokkal. Mit kell még megélnünk... Nehogy a posta is átvegye az e-mail-címeket, mint egyedi azonosítókat.

Éles balkanyar(intás)

Kevesen tudják tisztességesen lekanyarítani az at-jelet, vagyis a magyarul kukacnak becézett micsodát. Néha én is elfelejtem, pedig nagyon egyszerű: először egy írott kis „a" betűt formázunk, majd annak végvonalát meghosszabbítjuk a betű körül visszafelé, mintegy bekarikázva azt, és már készen is van a kukac. De mit kukacoskodok itt a cikk végén? Ez ma már általános iskolai tananyag, a programozók tudják, a @ az ABC nulladik, közvetlenül az A előtti tagja (az ASCII táblában). Statisztikailag előbb vagy utóbb a 10 leggyakrabban használt magyar karakter közé fogja felküzdeni magát. Ez hát a kukac karrierje!

Herczeg József

INFORMÁCIÓKÉRÉS: 0528 ▼

INFORMÁCIÓKÉRÉS: 0566 ▼

INFORMÁCIÓKÉRÉS: 0561 ▼

INFORMÁCIÓKÉRÉS: 0556 ▼

MQSeries

„Köztesware" technológia

Az MQSeries technológia lényege két fogalom: az üzenetküldés és az üzenetek sorbaállítása. Ezzel olyan programok, amelyek eltérő platformokon, eltérő nyelveken íródtak, konzisztens módon tudják megoldani az egymás közti kommunikációt, a hálózatos programozás bonyolultságának elkerülésével.

Manapság sok szó esik kliens/szerver rendszerekről. Az még csak hagyján, hogy nincs egyetértés a kifejezés írásmódjáról, de a fogalom definiálása is többféle. Ha valaki tőlem kérné, hogy adjak rá tömör meghatározást, akkor annál a nem túl sokat mondó, ám kevés támadási felületet nyújtó meghatározásnál maradnék, hogy olyan rendszerről van szó, ahol az alkalmazás szolgáltatáskérő és szolgáltatást nyújtó részekből áll össze. A teljesség kedvéért meg lehet még jegyezni, hogy a kliens és a szerver részeket összekötő (bikkfanyelven költött újabb szóval middleware) szoftverkomponensre is szükség van.

MQSeries, az igazi middleware

E „köztesware" felelős a kliens kéréseit megbízhatóan, gyorsan, biztonságosan, programozhatóan, olcsón, platformfüggetlenül és hordozhatóan megoldani. Egy-egy „köztesware" már korábban is felbukkant, az adatbáziskezelés területén ebbe a kategóriába sorolhatók az ODBC meghajtók. Ezek feladata, hogy elkülönítsék a felhasználó programját az őt adatokkal ellátó adatbáziskezelőtől. Ezt az SQL utasítások átadásával és az eredmény visszajuttatásával teszik meg.

E meghajtók azonban problémaspecifikus szoftverek, azaz közvetlenül nem segítenek abban, hogy az építendő rendszerben az adatok megosztott tárolása mellett az alkalmazás logikáját is optimálisan rendezzük el a felhasznált infrastruktúrában. (Ha kötözködni akarnék magammal, megjegyezném, hogy tárolt procedúrákkal és trigger programokkal bizonyos mértékig ez a megosztás is megoldható. Ám az mindenképp igaz, hogy ezek nem részei az alaplogikának, és egyéb, itt most nem részletezett okok miatt használatuk nem minden esetben javasolt.)

Az üzenetorientáltság lehetővé teszi, hogy a kommunikáció ne blokkolja le az egymással beszélgető programokat, mert nem közvetlenül hívják egymást. Így akár egy rendszeren belül is tervezhetünk kliens/szerver alkalmazásokat.

Az ilyen programtervezés előnye, hogy később a különböző modulok tetszőlegesen áthelyezhetők másik rendszerre, akár egy eltérő operációs rendszerre is.

Az üzenetek sorbaállításával, a várakozósorokkal (queue) oldja meg a rendszer az üzenetek tárolását. A várólistára kerülő üzenet egyik attribútuma a sürgősség. A szolgáltatást végző rutin a batch jellegű és az online feldolgozást végző programok kéréseit megkülönböztetheti és a sürgős kérésekre koncentrálhat.

Az MQSeries valóban „köztesware", vagyis platformfüggetlen, olcsó, könnyen programozható, megbízható és biztonságos. Vegyük sorra e tulajdonságáokat.

— **Platformfüggetlen.** Az MQSeries technológiát alapvetően két részre lehet bontani: létezik MQSeries kliens, amely nem rendelkezik saját erőforrásokkal, csak használhatja a szerver rendszerén lévő MQ sorkezelőt, és ott van persze maga a szerver. A kommunikációs folyamatot három rétegre bonthatjuk: az MQI-t használó kliens programra, az MQI-t kiszolgáló sorkezelőre (Queue-Manager), illetve arra a protokollra, amely a sorkezelők közti kommunikációt teszi lehetővé. Ezek után MQI-nak azt a programozható felületet nevezzük, amellyel használatba vehetjük az MQSeries objektumokat és szolgáltatásokat. (A mellékelt kis keretes anyagban megadott platformokon van MQ szerver.)

— **Olcsó.** Az árról elég annyi, hogy eltörpül azokhoz a szolgáltatásokhoz viszonyítva, amelyeket a technológia nyújt. Ezekkel a szolgáltatásokkal a hálózatos programozás sok útvesztőjét kerülhetjük el.

— **Könnyen programozható.** Az MQSeries környezetről általában elmondható, hogy könnyen átlátható és nem jelent újabb jelentős kihívást az amúgy nem egyszerű kliens/szerver fejlesztés során. A szoftver két alkotórészre bontható: egyrészt arra az eszközkészletre, amellyel a queue-kat és a hozzájuk tartozó egyéb definíciókat kezelhetjük (nem feledkezve meg az autorizációs parancsokról), másrészt pedig egy programozói felületből, amely a következő hívásokból áll:

Az API neve — és funkciója

MQCONN, MQDISC — MQSeries menedzserrel kapcsolat létrehozása és lebontása.

MQOPEN, MQCLOSE — Queue megnyitása és bezárása, a fájlkezeléshez hasonlóan minden queue-t használat előtt meg kell nyitni. Ez alól csak a következő hívás jelent kivételt.

MQPUT1 — Ez a hívás egyben végzi el az adott queue megnyitását, az ebbe való írást, és a queue bezárását.

MQPUT — Megnyitott queue-ba írja az üzenetet.

MQGET — Megnyitott queue-ból olvas.

MQINQ, MQSET — MQSeries objektumok attribútumainak lekérdezése, és ezen attribútumok írása.

MQCMIT, MQBACK — Üzenetküldések koordinálása.

A programozói felület a hívások számát tekintve egyszerűnek mondható, az MQI sikeres használata nem valami technikai varázslattól, hanem a megfelelő tervezéstől függ.

— **Megbízható.** Az MQM alapelve a garantált üzenettovábbítás. Ezért ha a célrendszer éppen nem üzemel, vagy a kapcsolattal van gond, az üzenetek akkor is garantáltan elérik céljukat. Ez a tulajdonság rendkívül hasznos lehet olyan alkalmazások esetén, ahol például ügynökök offline módon végzik

Ahol van MQ szerver

IBM platformokon	Nem IBM platformokon
OS/400	HP-UX
MVS/ESA – CICS, IMS, TSO	SCO UNIX
AIX	Digital VMS VAX
OS/2	Tandem Guardian
	UnixWare
	Windows NT, 95
	DOS, Windows 3.x
	Sinix
	Sun SunOs
	Sun Solaris

munkájuk nagy részét, és csak időnként kapcsolódnak a vállalat szerverére. A felvett rendelések a kapcsolat felépültével automatikusan eljutnak a szerverhez.

— **Biztonságos.** Az MQSeries technológia lehetővé teszi az üzenetek titkosítását, és az operációs rendszertől függően az alap MQSeries objektumok is megvédhetők az illetéktelenekkel szemben.

A „batch window" probléma

A probléma lényege (egy áruházlánc példáján) a következő. Az eladások a napi forgalmi állományban gyűlnek, este ezek feldolgozásra kerülnek, és különféle feladások készülnek belőlük. A forgalom növekedésével a feladások egyre hosszabb időt vesznek igénybe, egyre nagyobb „vasak" vásárlásával oldják meg a cégek a problémát, csak azért, mert rendszerük nem megfelelően strukturált.

Az MQSeries megoldást jelent ezekre a gondokra, hiszen a klasszikus postázó/feldolgozó programokat átalakíthatjuk úgy, hogy ezek a napi munkával párhuzamosan végzik a háttérfeldolgozást. Így napközben alacsonyabb prioritással futva nem zavarják a tömeges online feldolgozást végző felhasználót, és kihasználhatják az üresjáratokat (például az ebédszünetet). Az online programok egyszerre rögzítik a tranzakciókat, és küldhetik a feladásokat a feladatok szerint felosztott várakozósorokra.

Az MQSeries megoldhat néhány komoly számítástechnikai problémát. Könnyen építhetünk olyan alkalmazásokat, amelyek optimálisan használják fel a rendelkezésre álló számítástechnikai eszközparkot, PC-ink a legújabb Windows operációs rendszerekkel vagy OS/2-vel, szép grafikus (OS/2 esetében objektumorientált) felületükkel kliensekké válhatnak, és nekik megfelelő módon alkalmazás- vagy adatbázisszerverként használhatjuk az erre a feladatra legalkalmasabb AS/400-as vagy RS/6000-es rendszereket.

Cikksorozatunk következő részében egy olyan élő alkalmazást mutatunk be, amely AS/400-ról a rendszer biztonsági eseményeit rögzítő naplóból küldi át a fontosnak ítélt eseményeket egy OS/2-n futó szerverprogramnak, amely a rekordot egy Lotus Notes formként későbbi lekérdezésre rögzíti. A technikát más területen is használhatjuk, például a tranzakciós rendszer és a csoportos munkát, munkafolyamat-irányítást támogató Lotus Notes összekötésére. Ezt követően bemutatunk alapvető MQSeries technikákat és egy egyszerű MQSeries-re alapuló tranzakciókezelő rendszert.

A következő cikk megjelenéséig egy kis „muníció", amely segítheti a kitartó olvasókat, hogy kipróbálják a mostani CD mellékleten lévő demó MQSeries kódot. Installálása egyszerű, így azt itt nem is szükséges kifejteni, legfeljebb arra hívom fel a figyelmet, hogy az NT és Unix rendszert használók hozzanak létre egy „mqm" nevű klienset.

A demóprogram, amellyel az MQSeries kipróbálható, ún. „triggering" technikát használ. (Talán jobb, ha nem fordítom le a „triggering" kifejezést elsülős technikának.) A módszer lényege, hogy egy kliens program kéréseket küldhet egy alkalmazási „queue"-ba. Az alkalmazási „queue"-ban nincs semmi különös, ő is egy a sok közül, kivéve azt, hogy számára megadtuk a TRIGGER attribútumot, amely jelzi a sorkezelőnek (a QM-nek), hogy bizonyos feltételek esetén küldjön egy másik sorra egy „trigger" üzenetet. Ez az üzenet igazából nem más, mint egy programleírás. Az alkalmazás várakozósorának fontos attribútumai így a következők:

TRIGGER, vagy nem TRIGGER várakozósor.

TRIGTYPE, mi az az esemény, ami egy „trigger" üzenetet kivált, az első a várakozósoron megjelenő üzenet, vagy egy bizonyos számú üzenet megléte.

PROCESS, én egyszerűen egy programleírásnak nevezném, ennek itt csak a neve tárolódik.

INITQNAME, inicializáló várakozósor, erre a várakozósorra figyel az ún. trigger monitor.

Az üzenetküldözgetés sokszögének következő objektuma, maga a program (PROCESS) leírása. Ez többek között a program/tranzakció nevéből, futási paraméteréből, típusából áll.

Tehát ott tartunk, hogy megjelenik egy üzenet az alkalmazás várakozósorában, erre a QM küld egy üzenetet az INITQNAME-ben meghatározott várakozósorra. Ahhoz, hogy a kérő (kliens) alkalmazásunkat valaki ténylegesen kiszolgálja, valakinek el kell indítania a szolgáltatást végző programot is. Ezt a feladatot végzi az ún. „trigger monitor". Ez a program figyeli az INITQNAME-ben szereplő várakozósort, egyenként leszedi az ott megjelenő MQTM-trigger típusú üzeneteket, és a bennük szereplő programokat/tranzakciókat elindítja. Ilyen programot mi is írhatunk, de OS/2 alatt ezt elvégezheti a „runmqtrm" nevű „EXE".

A megadott program nagyon egyszerű funkciót lát el, de csekély módosítással könnyen alkalmassá tehető mindenféle kliens/szerver feladat elvégzésére. A feladat a következő: a kliens program (AMQSREQ, AS/400-on AMQSREQ4) egy szövegfájlból üzeneteket küld a paraméterként megadott várakozósorra. Ezt az üzenetet fogja a trigger monitor által elindított program (AMQSECHA, AS/400-on is) visszaírni a kért válasz-várakozósorba (SYSTEM.SAMPLE.REPLY). Ez a várakozósor OS/2-n dinamikus sorként van kialakítva. A trigger monitor neve „AMQSTRG", AS/400-on használhatjuk az „AMQSERV4"-et. A szükséges objektumok a következők:

Kliens program: AMQSREQ, AMQSREQ4
Alkalmazás várakozósora: SYSTEM.SAMPLE.ECHO
Trigger várakozó-sor: SYSTEM.SAMPLE.TRIGGER
Válaszsor: SYSTEM.SAMPLE.REPLY
PROCESS definíció: ECHO

A futtatáshoz a következő lépéssort hajtsuk végre:

1. Hozzunk létre egy Queue-Managert:

CRTMQM F400.TESZT.MQM

A létrehozásnál javasoljuk a következő konvenciót:

Vállalat_neve.Szerver_neve.MQM

Később a sorkezelő neve helyére mindig a saját sorkezelőnk nevét helyettesítsük be.

2. Indítsuk el a QM-et.

STRMQM F400.TESZT.MQM

3. Hozzuk létre a szükséges rendszerobjektumokat (alapértelmezésként szolgáló várakozósorok és egyebek).

RUNMQSC F400.TESZT.MQM < C:\MQM\MQSC\AMQSCOMA.TST > EREDM1.KI

4. Hozzuk létre az mintaalkalmazáshoz szükséges objektumokat:

RUNMQSC F400.TESZT.MQM < C:\MQM\MQSC\AMQSCOS0.TST > EREDM2.KI

5. Egyik ablakban indítsuk el az \MQM\TOOLS\C\SAMPLES\BIN könyvtárból a következő parancsot:

AMQSTRG SYSTEM.SAMPLE.TRIGGER F400.TESZT.MQM

Ez a trigger monitorunk.

6. Másik ablakban indítsuk el az \MQM\TOOLS\C\SAMPLES\BIN könyvtárból a következő parancsot:

AMQSREQ SYSTEM.SAMPLE.ECHO F400.TESZT.MQM < text.fil

Ez a kliens programunk. Futtatása előtt hozzuk létre a text.fil fájlt, és írjunk bele néhány sort.

Ha minden jól ment, és sorkezelőnk nevét és objektumait megfelelelő kis- és nagybetű-kombinációval írtuk, a kliens program kiírja a válaszként kapott sorokat. Mindenkinek sok sikert és örömet a programozáshoz. Kérdésekkel az f400@pronet.hu címhez lehet fordulni.

Molnár Balázs

Az ötödik...

A változatosság gyönyörködtet

Eredetileg nem úgy terveztük, hogy mostani számunkba feltétlenül kell extra CD-ROM melléklet, de azután arra gondoltunk, hogy miért ne legyen a jubileum még kerekebb: 15. évfolyam, 150. szám, 5. CD-ROM...

Előre bocsátjuk, hogy egy lényeges kérdés az ötödik Új Alaplap-CD elkészítése után is nyitva maradt. Nevezetesen, hogy állandó részévé váljék-e ez a melléklet-forma az Új Alaplapnak, vagy sem.

Legutóbbi számunk véleménykutató kérdőívén azért is intéztünk olvasóinkhoz kiemelt kérdést a CD-ROM melléklet ügyében, mert a válaszok összesítését mintegy „népszavazásnak" szeretnénk tekinteni. Az elsőként visszaérkezett néhány száz kérdőív tanúsága szerint azonban a CD-vel kapcsolatos vélemények igencsak megoszlanak, így félő, hogy később sem kerekedik ki belőlük túlnyomó többséget kitevő „igen" vagy „nem", és akkor megint szerkesztőségünkre hárul a döntés terhe, hogy áttérjünk-e az állandó CD-re, vagy maradjon a jelenlegi floppys (és mellé alkalmanként extra CD-s) megoldás.

Az új futtatórendszer is erre az alkalomra készült, továbbra is kísérleti jellegűnek tekintjük, használatának „bonyolultsága" a floppymellékletét idézi. Jelentős változás, hogy szakítottunk a DOS-ból (is) történő indítással, mert szinte mindegyik alkalmazás Windows-felületet igényel, és így egyenesen botorság lett volna ügyes csellel kifogni a Windowson, hogy „csakazértis DOS-ból" jussunk el hozzá. Az egyetlen DOS-os alkalmazásnak — a LÁ.VA Manager Iroda „IQtató" programjának — a windowsos megfelelője szintén rajta van a CD-n.

A Delphiben megírt futtatórendszer semmiféle extrával nem terheli a felhasználók gépének Windows rendszerét, minden erőforrást az eredeti beállításoknak megfelelően kezel.

A futtatható programok minimálisan az alábbi operációs rendszert igénylik. (A Windows 3.x esetében persze több olyan is van, amely „kedveli" a hagyományos Windows 3.x rendszer 32 bites kiegészítését.)

DOS, Windows 3.x:
— Titkársági IQtató
— Epson meghajtók
Windows 3.x:
— Platinum szoftverek
— IBM MQSeries middleware
— MS Office 97 újdonságok
— Merlin-bemutató
— Visual DataFlex-bemutató
— Crimp for Windows
— Britannica CD 2.0 demó
— Alarmix Hungary weblap
— Útmutató a szabványokhoz
— Profi-Média nyelvoktató
Windows 95:
— DirectX driver
— Neverhood játék
Windows 95, Windows NT:
— Allegro multimédia-bemutató

Mint látható, a CD-ROM tematikája elég változatos, néhány „high end" szoftver kipróbálási verziójától az interaktív és autodemókon meg drivereken át egészen a multimédiáig és a játékig. Célunk az volt, hogy a változatosság és hasznosság érvényesüljön, a kínálat „ne feküdje meg senkinek a gyomrát". Külön szeretnénk felhívni a figyelmet egy „hidden" állományra. A magyarul éppen most debütáló Office 97 dokumentumegyüttesének viewer programja véleményünk szerint előkelő helyre kerülhet a segédprogramok között, mert amíg valaki nem tér át az új változatra, ezzel „fogyasztani" tudja a Word 97-es fájlokat is.

Azok az olvasóink, akik beleszerettek az 1996. áprilisi CD-nken közreadott Warpba, rendszeresen kérték tőlünk a Merlin végső béta-változatát is. Nálunk azonban most „csak" egy élvezetes interaktív Merlin-demó szerepel, mert az OS/2 Times áprilisi CD-mellékletén a teljesértékű Merlin-béta már forgalomba került.

A mi CD-nk a mélyebb szakmai ismeretekkel bíróknak kínál viszont egy igazi csemegét, egy „nagy fogást": a Platinum szoftveregyüttest. Az adatbáziskezelők és adatbázis-alkalmazások biztonságos, hatékony üzemeltetéséhez ajánlott termékcsalád megtekinthető az Ifabón is, az IQSoft standján. Az Új Alaplap CD-jén közreadott Platinum-verzió pedig az IQSoft-tól kapható regisztrációs számmal feléleszthető, időkorlátosan teljes funkcionalitásában kipróbálható. (Ha aranylemez vagy platinalemez nem is lesz az Új Alaplap mostani CD-mellékletéből, Platinumlemezként esetleg emlegetni fogják.)

A CD-n elhelyezett játék, az igen szellemes és látványos Neverhood-demó, valamint az Allegro-Scala multimédia-bemutató indítása gondot okozhat olyan Win95-környezetben, ahol nincs telepítve DirectX-driver. Mi elhelyeztük a CD-n a DirectX driverét, de javasoljuk, hogy telepítésére csak az vállalkozzék, aki megfelelő rendszerismerettel bír, ennek hiányában a drivertelepítést bízza inkább szakemberre. A DirectX-vezérlő ugyanis elég hiányosan van dokumentálva, és „fiatalsága" folytán inkompatibilitására, összeférhetetlenségére a sokféle rendszerkörnyezetben még nem nagyon derülhetett fény. A tesztelés során ezt a veszélyt magunk is tapasztaltuk, ezért nem tettük a drivert közvetlenül indíthatóvá.

Eddigi CD-mellékleteinkhez hasonlóan ez a CD-ROM sem a lap állandó tartozéka, hanem a floppymellékleten felüli *extra,* következésképpen extra költségvetésből kellett gondoskodnunk finanszírozásáról, márcsak azért is, hogy utcai árusításban a lap ne kerüljön a mostanra megemelt összegnél többe (miközben az előfizetők többletköltség nélkül kapják meg ezt a számot is). Éppen ezért az IBM Magyarországot, az IQSoft Rt-t, a Microsoft Magyarországot, az Allegro Bt-t, a Profi-Média Kft-t, a Next Software Kft-t, az Alarmix Kft-t, valamint a LÁ.VA Manager Irodát szakmai közreműködésükön felül vállalt anyagi hozzájárulásukért külön köszönet illeti.

A CD-melléklet munkálataiban való közreműködéséért köszönetet mondunk továbbá Simay Endre Istvánnak, az új futtatórendszer készítőjének, valamint Horlai Jánosnak, aki a rendszerszintű kérdésekben volt segítségünkre.

Varga János

AST CD-ROM

Az AST számítógépek hazai disztribútoránál, a G70 Kft-nél elkészült az első AST-Support CD-ROM. Ennek fő attrakciója az AST Pronto Pro nevű program, amely fel van készítve a jelenlegi, illetve néhány éves modellek összes adatával (konfigurációk, specifikációk, alkatrészleírások, hibaleírások és hibamegoldások, driverek, segédprogramok stb.). A CD-n található még AST-demóprogram, magyar Powerpoint-prezentáció és egy .AVI fájl. A CD-ROM-ot bárki megkaphatja, csak jeleznie kell igényét a G70-nél.

Magyar Office 97

Vélhetően az Ifabo egyik szenzációja, s az utána következő hetek egyik slágerterméke lesz a népszerű alkalmazásegyüttes magyar változata. Az MS Office 97 Pro magyar változatának bevezető ára például mindössze 69 900 forint (upgrade-ként 39 000 forint) a KimSoftnál.

Világrekord

A Digital számítógépei már korábban is jó barátságban voltak az SAP R/3 integrált vállalatirányítási alkalmazással, és az a rekord, amelyről lapzártakor kaptunk hírt, ezt még inkább megerősíti. Szoftveroldalon SAP R/3 release 3.0E, Digital Unix V4.0B és az Oracle 7.3.2.3 adatbáziskezelő alkotta a rendszert, amely a Sales & Distribution modul teljesítményét mérő benchmarkban világrekordnak minősülő eredményt produkált. Hardveroldalon 10 db AlphaServer 8400-as szolgálta ki az alkalmazásokat. Az eredmény értékét emeli, hogy a lehetséges 12 processzorból csak 10-et használt az adatbázis-szerver. (Mellesleg 10 GB memóriával és 284 GB háttértárkapacitással üzemelt a rendszer.) A rekord-benchmark adatai számszerűen: átlagosan 0,42 másodperces válaszidők, miközben egyidejűleg 2000 benchmark-felhasználót kellett kiszolgálni, akik óránként 691 000 dialóguslépést hajtottak végre. Ez a teljesítmény óránként 230 000 (üzletileg teljesen feldolgozott) rendeléssornak felel meg.

Synergon az Ifabón

Először mutatkozik be Synergon Informatikai Rt. néven a nagyközönség előtt is az a vállalat, amely — mint ismeretes — az Optotrans és a Rolitron Informatika egyesülése révén jött létre a közelmúltban, és ez évi 4 milliárd forintos tervezett forgalmával máris a legnagyobb magyar rendszerintegrátornak tudhatja magát. Néhány ifabós kiállítási terület, amely jól jellemzi azt a komplett termék- és szolgáltatásválasztékot, amely az új cég sajátja: integrált vállalatirányítás (JD Edwards, Prism), kommunikációs rendszerek (Cisco, Newbridge, Sun, Lanoptics, Cabletron), hálózatok (BICC, Systimax), intelligens épületek kommunikációs rendszerei, telefonközpontok, alközpontok, integrált ügyviteli szoftvermegoldások (Lotus Notes, Microsoft-alapú fejlesztések), szoftverauditálás.

9001-es X-Byte

Az X-Byte 12 éve épít adatátviteli hálózatokat országszerte, s nemcsak mint hálózatépítő cég első az országban — „mellesleg" élettartam-garanciát vállalva az épített hálózatokra —, hanem saját szakterületén az első olyan cég, amely megszerezte az ISO 9001-es minősítést. Ez voltaképpen a kialakult gyakorlat elismerése, hiszen élettartam-garanciát csak olyan esetben szabad vállalni, ha a felhasznált anyagok és eszközök minősége minden szempontból kiváló (amit ISO 9001 minőségbiztosítási rendszer megkövetel), és a munkavégzés is ennek megfelelő. Az elmúlt tizenkét év alatt több mint 2000 ügyfele volt a cégnek, és ez összesen több mint 5000 általuk épített hálózatot jelent. Gyakorlatilag valamennyi a referencialistát gyarapítja, hiszen reklamáció csak elvétve akadt.

A lézernyaláb „kezelése"

Alighanem a tavaly ősszel még csak prototípusként bemutatott, idén viszont már sorozatban gyártott LaserBit lézeres optikai adatátviteli rendszer lesz a CrownTech Ifabo-bemutatójának slágere. A látogatók működés közben tekinthetik meg a kiállításon a rendszert, s akár a kezükkel is minden veszély nélkül megszakíthatják a lézernyaláb útját. Az új termékek között található az NBase GigaHub Switch is, amely 16 Mbit/s-os sávszélességű.

Víruskereső az IBM-től

Sokakat bizonyára meglep a hír, hogy az IBM a víruskeresők területén is babérokra törekszik. Az új IBM AntiVirus 2.5.2. azért érdemel külön figyelmet, mert a legelső olyan szoftver, amely képes feldolgozni az új Office 97 formátumait, hogy a digitális kártevők gyorsan és megbízhatóan felfedezhetők és semlegesíthetők legyenek. „Bár maga az Office 97 is megtalál néhány ismert makróvírust, biztos védelmet csak az IBM AntiVirus képes nyújtani az új típusú makróvírusok — és még vagy 8000 egyéb társuk — ellen" — állítja az IBM közép-európai kereskedelmi igazgatója. A Virus Bulletin című lap szerint egyébként az IBM Antivirus 2.5.2 három és félszer gyorsabb a Norton AntiVirusnál, a MacAfee VirusScannél pedig nyolcszor gyorsabb működés jellemzi. Egyfelhasználós ára átszámítva mintegy 10 000 forint.

600 MHz-es Alpha

Elkészült a világ leggyorsabb mikroprocesszora, a Digital Alpha 21164-es. A 600 MHz-es processzor másodpercenként 2,4 milliárd utasítás végrehajtására képes, 18 SPECint95 és 27 SPECfp95 becsült számítási kapacitás jellemzi. Az első, ilyen processzorral szerelhető alaplapok (AlphaPC 164LX) 1997 nyarán lesznek megvásárolhatók.

300 MHz-es Alma

Az Apple bejelentése szerint a Power Macintosh 6500/300-as gép az első 300 MHz-es rendszer a mai piacon. A Power Macintosh családot egy új Motorola PowerPC chip hajtja, és a 6500-as családot jelentősen tovább javították a multimédiás lehetőségek, a DTP, a 2D és 3D grafikai képességek területén. Cégen belüli tesztek alapján — amelyek során létező alkalmazásokat futtattak — megállapították, hogy az új Power Macintosh rendszerek kétszer akkora sebességet tudnak produkálni, mint az MMX technológiájú 200 MHz-es Pentium processzorok. A 6500-as rendszerek Magyarországra érkezése idén nyárra várható.

50 éves a tranzisztor

Idén 50 éves a Bell Laboratories három Nobel-díjasának nevével fémjelzett találmány: az első tranzisztor germániumba préselt két aranylemezből álló szerkezet volt, és amikor erre feszültséget kapcsoltak, azt tapasztalták, hogy erősítőként működik. Mindössze három érdekesség a Lucent Technologies által az évforduló alkalmával kiadott sajtóközleményből: ha tranzisztorok helyett elektroncsövekkel kellene elkészíteni egy mai notebook számítógépet, pusztán a központi tárolójához szükséges tápellátó rendszerek, hűtőberendezések, vezetékek és tartókeretek megtöltenék a New York-i World Trade Center egyik (mintegy 400 méter magas) épületét; a világon ma kb. 200 milliárd tranzisztor van, így (elméletileg) minden emberre egyenként jut vagy 60 millió darab; az elkövetkező évtizedekben a mikroáramkörök egy körömnyi méretű lapkán 1 milliárd tranzisztort fognak tartalmazni.

APRÓHIRDETÉSI MEGRENDELŐLAP

Kérem, hogy az Új Alaplap következő számának Mikrobazár rovatában közöljék az alábbi szövegű apróhirdetést:

....

....

....

....

(Maximális terjedelem: 300 betűhely)

Alaplap Posta

MEGRENDELÉS

Az Új Alaplap 1997/5. számának 67. oldalán ismertetett könyvek közül postai utánvétellel megrendelem az alábbiakat.

Név: ....

(Cég:) ....

Cím: ....

Helység: ....

Irányítószám: ....

Az árat a küldemény átvételekor a kézbesítési díjjal együtt kifizetem.

Dátum:

/aláírás/

Előfizetés az Új Alaplapra

Az 1997/......... számtól kezdődően előfizetem az Új Alaplap című, havi számítástechnikai folyóiratot ...... példányban, ❑ 1 évre, ❑ 1/2 évre.

Az éves előfizetési díj 4356,– forint.

Az előfizetési díj kiegyenlítéséhez:

❑ **Számlát** kérek (banki átutalással fizetek).

❑ **Átutalási postautalványt** kérek.

Név: ....

(Cég:) ....

Cím: ....

Helység: ....

Irányítószám: ....

Dátum: ....

/aláírás/

INFORMÁCIÓKÉRÉS

Az Új Alaplap 1997. májusi számának hirdetéseihez

Kérem, hogy az itt általam **BEKARIKÁZOTT KÓDSZÁMÚ** hirdetésekkel kapcsolatban küldjenek részemre bővebb tájékoztatást.

Beküldhető: 1997. május 31-ig

0501	0519	0537	0555
0502	0520	0538	0556
0503	0521	0539	0557
0504	0522	0540	0558
0505	0523	0541	0559
0506	0524	0542	0560
0507	0525	0543	0561
0508	0526	0544	0562
0509	0527	0545	0563
0510	0528	0546	0564
0511	0529	0547	0565
0512	0530	0548	0566
0513	0531	0549	0567
0514	0532	0550	0568
0515	0533	0551	0569
0516	0534	0552	0570
0517	0535	0553	0571
0518	0536	0554	0572

FELADÓ **Feladáskor kérjük bérmentesíteni!**

A) Egyéni érdeklődő:

Név: ....

Cím: ....

Helység: ....

Irányítószám: ....

B) Vállalati érdeklődő:

Cég: ....

Ügyintéző: ....

Cím: ....

Helység: ....

Irányítószám: ....

Telefon/Fax: ....

Bélyeg helye

Új Alaplap
szerkesztősége
I., Márvány u. 17.
Pf. 571
Budapest 1539

Belföldön díjmentesen feladható

Új Alaplap
szerkesztősége
I., Márvány u. 17.
Pf. 571
Budapest 1539

Minden PC-hez kell egy jó alaplap!

És egy Új Alaplap!

Belföldön díjmentesen feladható

Új Alaplap
szerkesztősége
I., Márvány u. 17.
Pf. 571
Budapest 1539

FELADÓ: **Feladáskor kérjük bérmentesíteni!**

Név: ....

Cím: ....

Helység: ....

Irányítószám: ....

Telefon: ....

☐ A hirdetés egyéni és egyedi jellegű, ezért kérem ingyenes megjelentetését. Kijelentem, hogy annak tartalma nem sérti senki szerzői jogát.

☐ A hirdetés kereskedelmi célt szolgál. Mellékelem a soronként (60 karakterenként) 300 forintnak megfelelő összeg átutalásáról az igazoló szelvény másolatát.
A címzett: Új Alaplap, 1539 Budapest, Pf. 571, illetve átutalásnál az OTP 11701004-20171649 számlaszám.

Bélyeg helye

Új Alaplap
szerkesztősége
I., Márvány u. 17.
Pf. 571
Budapest 1539

Az Új Alaplap lemezmellékletének tartalma — 1997. MÁJUS

Szabványközpontú Exchange Server

A Microsoft Exchange Server új, 5.0-s verziójának legfőbb tulajdonsága, hogy magába foglalja a legfontosabb Internet szabványokat, miközben továbbra is támogatja a meglévő levelező és csoportmunka-segítő technológiai szabványokra épülő megoldásokat. A levelezőcsomag új változata a korábbinál egyszerűbb telepítésével és felügyeletével is kitűnik; eleve úgy tervezték, hogy valamennyi fontos Internet protokollt (SMTP, POP3, NNTP, LDAP, HTTP, HTML, SSL) és a legfontosabb e-mail-szabványokat (X.400, X.500, MAPI) is támogassa. A termék nem kevesebb mint 21 új funkcióval, jellemzővel rendelkezik. Ezek közül a Microsoft Outlook 97 desktop információmenedzser az Exchange Client és a Schedule+ továbbfejlesztése Win32-alapú desktop géphasználók részére. Az Exchange Active Server Components JavaScriptet használ arra, hogy a felhasználók MS Internet Explorer, Netscape Navigator vagy más böngészőket futtatva „élő" írási/olvasási kapcsolatban legyenek az Exchange szerverekkel. Az Internet Mail Service Wizard (IMS) a rendszeradminisztrátoroknak könnyíti meg az IMS konfigurálását, ennek következtében SMTP protokoll alapon az Interneten keresztüli e-mail üzenetek küldését és fogadását. Újdonság az Exchange 5.0-ban a Lotus cc:mail-lel, a Novell GroupWise-zal, a Collabre Share környezettel és az Apple Macintosh-felhasználókkal csoportütemezésben való együttműködés képessége.

Fast Ethernet piaci csata

A kaliforniai Dell Oro Group piacelemző cég előrejelzése szerint a kapcsolt Fast Ethernet hálózati eszközök és megoldások piaci forgalma 1997-ben eléri a kétmilliárd dollárt, 2000-re pedig meghaladhatja a 6,5 milliárdot. A felhasználókért az iparág három multija, a Bay Networks, a Cisco és a 3Com van versenyben, néhány egyéb, de különleges igényt kielégítő más szállítóval együtt. A Dell Oro szerint a csata jelenlegi állása szerint a 3Com vezet: az asztali kapcsolástechnológiára vonatkozó legújabb Dell Oro-adatok szerint a 3Com részesedése 46,1%, a Cisco 18,2%-kal áll a második helyen. A munkacsoportos Fast Ethernet hubok szegmensében a 3Com 34,5%-ot szerzett meg, míg legnagyobb riválisa, a Bay Networks 16,6%-ot. A 3Com nem régi bejelentése szerint a Fast Ethernet hálózati adapterkártyák (NIC) terén 36%-kal vezető pozíciót szerzett, nem kis mértékben annak köszönhetően, hogy a cég jelentős árcsökkentéseket hajtott végre a rétegelhető, méretezhető SuperStack II. és a fixen beépített Oncore moduljára.

Novell: akció itthon és a világban

A Novell Magyarország április 4-én bejelentette, kéthónapos akciót kezdett az IntranetWare for Small Businesses (ISB) termékkel, megcélozva a hazai kisvállalkozásokat, igen kedvező beszerzési lehetőségeket ajánlva nekik. Akciós ajándékként minden vásárlónak három hónap ingyenes Internet-használatot és az akció idején ingyenes hotline-támogatást biztosítanak. Árukapcsolásként július 31-ig minden ISB-vásárló ingyenesen megkap egy Cheyenne programtermékeket tartalmazó CD-t, amely a kisvállalatok számára további hálózati szolgáltatásokat nyújt (JetServe 3.3 biztonsági mentőrendszer; InocuLAN hálózatos axxx megoldás; FaxServe — kétfelhasználós központi faxoló; ArcServe archiváló program). Webes kiterjesztések jellemzik a legújabb Novell—Oracle együttműködési szerződést. A megállapodás részeként a Novell vállalta, hogy integrálja és együtt árulja az új Oracle Web Application Server 3.0-at az IntranetWare-rel, Web Serverrel és NDS-sel. Az Oracle pedig a technológiai integráció központjául szolgáló, közös marketing-eladás, oktatás, támogatás elősegítésére új részleget, a Novell Products Divisiont hozza létre. A Novell és a Netscape stratégiai megállapodást írtak alá a jelen és jövő Novell platformjain működő intranet és extranet alkalmazások közös fejlesztésére és piaci elterjesztésére. A szerződő cégek Novonyx néven olyan új közös vállalatot hoztak létre, amelynek első feladata, hogy a Netscape SuiteSpot szerverszoftver készletét és a Novell IntranetWare programcsomagot egymáshoz integrálja és árusítsa. A Novell és a Sun Microsystems ismertette, közösen jelentették be a San Francisco-i JavaOne Java fejlesztői konferencián, hogy stratégiai partnerkapcsolatukat kiterjesztik Internet és vállalati intranetek platformfüggetlen megoldásainak szállítására. A két cég arra vállalkozik, hogy integrálja a Novell hálózati szoftvertechnológiát a Sun „Write Once, Run Anywhere" Java technológiájával. Ezáltal könnyen előállítható, biztonságosan elérhető, menedzselhető és a vállalat egész területén elosztható Java alkalmazások fejleszthetők.

Lucent: gigabites korszakba lépve

Újabb mérföldkövéhez érkezett a kábelezés terén mindig is innovatív, piacvezető technológiákat kidolgozó amerikai Lucent Technologies cég: bejelentette Systimax Gigaspeed nevű új, rézvezetős kábelét, amely kifejezetten a megbízható, nagy sebességű (max. 1 Gbit/s) adatátvitelre lett kifejlesztve. „Az új UTP kábel felette áll a jelenlegi szabványkategóriáknak" — jelentette ki T. C. Tan, a Lucent fejlesztőintézetének a Bell Labs Privát Hálózatoknak a vezetője. A Gigaspeed a szabványokban előírtaknál jóval kedvezőbb jelminőséget, jelerősséget és jelkiegyensúlyozottságot eredményez, miközben csökkentették a kábelkisugárzás mértékét és tovább minimalizálták az áthallást. Olyan új, a nagyobb frekvenciák (550 MHz) átvitelére is alkalmas kábel kerül piacra, amelyet sok más, alacsonyabb minőségű kábel nem képes teljesíteni. Az új kábel lehetőséget teremt nagy sávszélességigényű alkalmazások (pl. interaktív multimédia, Fast Ethernet, 1,2 Gbit/s-os ATM) kiszolgálására, velük párhuzamosan az egyéb (adat, hang, video) jelek, valamint szélessávú analóg jelek átvitelére. A Lucent ma az egyetlen kábelgyártó, amely az 550 MHz-es szélessávú videoalkalmazásokat kiszolgáló terméket állít elő.

HP: Java-képes Netstation

A Hewlett-Packard, amely korábban kevesebb figyelmet fordított az Internetre és az NC technológiára, március 31-én váratlanul bejelentette saját hálózati számítógépét, a HP NetStationt, mégpedig rögtön két modellben, a HP Entria II és Envizex II X terminálokra épülve. Az Internet/intranet hálózatokon való kommunikációra és a — még nem helyi, hanem szerver alapú — Java-alkalmazások, valamint „fürtös" hírcsoport és e-mail-alkalmazások futtatására képes NetStation támogatja a Netscape Navigator 3.0-ra alapozott, a szerveren futó Navio Navigatort, és gyártója szerint egyike a piacon kapható legolcsóbb grafikus desktop gépeknek. A HP idén őszre ígéri, hogy a Netstation teljes értékű hálózati számítógép lesz, a Windows, Unix, NT és más fontos környezetek támogatásával. A Navio Navigator használata is idén őszre lesz megoldva. A belépés szintű HP Netstation (HP Entria II-re alapozva) ára az USA-ban 700 dollár lesz. Az Entria II 120 MHz-es rendszer, amely dinamikus hostgép-konfigurálási protokolljának megfelelőjével, grafikus start-up képernyővel, távoli konfigurálási képességekkel, dinamikus billentyűzetkiosztással és plug and play képességekkel rendelkezik. A 133 MHz-es Envizex II munkaállomás stílusú, 16 bites grafikát, dupla puffertárolást támogat. Ezáltal komplex grafikák, 3D-s képek megjelenítésére van lehetőség. Az Envizex a 100Base-T és 100VG-AnyLAN nagysebességű hálózati protokollokat kielégítve, nagy sebességű hálózati használatra alkalmas. Az új Netstation-nel (az IBM is így nevezi hasonló célú eszközeit — a szerk.) potenciális felhasználóként a HP elsősorban a jelenlegi ASCII-, 3270- és/vagy X-terminál-használókat célozza meg.

Kovács Attila

Kéz a kézben

Heterogén hálózatok

Hogyan történik a Microsoft NT szerver és kliens menedzselése NetWare hálózatban, novelles eszközökkel? Ez a téma napjainkban egyre aktuálisabb lesz.

A Novell a piac új kihívásaira válaszul termékeiben magasabb szintre emeli a Windows NT szerver és kliens támogatását, ezzel kívánva érzékeltetni, miként látja elképzelhetőnek a két rendszer egymás mellett élését. Az IntranetWare kliens (Workstation Managerrel), a Novell AppLauncher, az Administrator és a Mobile Services, valamint a ManageWise kliens és SNMP ügynök révén a Novell egy komplex megoldást nyújt heterogén hálózatok használatára és hatékony menedzselésére.

Szintén érdekes, hogy az NT-s biztonsági és könyvtárszolgáltatást már két cég is megvalósítja a jövőben: a Novell Directory Services és az IBM a Directory and Security Services-zel, ez utóbbi ráadásul több platformon meglévő termék.

Bár a Novell lépéséveel szükségszerűen közeledik a Microsofthoz, mégis inkább saját termékét tartja jobbnak. A Novell-stratégia a Windows NT által megvalósítottól leginkább a rugalmasságban és a kezelhetőségben különbözik, a szabad konfigurálhatóságot helyezve előtérbe. Bár az NT Zero Administration-je jelentős lépés ebben az irányban, elemzők szerint még mindig elmarad a Novell adminisztrációtól, csakúgy, mint az IBM-től, amely a Netfinity termékcsaládot kínálja a Warp Serverben.

A Novell Directory Services ezen felül szintén nagymértékben csökkenti az adminisztrációs költségeket, így a rendszergazda több időt tölthet a végfelhasználók problémáival, ami igen fontos tényező lehet.

IntranetWare for NT

Az IntranetWare Windows NT kliense mindent tud, amit az IntranetWare nyújt, ezáltal tökéletesen integrálja a Windows NT Workstationt a hálózatba. A szoftver a helyi NT Workstation szoftverhez csatlakozik, hogy leegyszerűsítse a bejelentkezést. Egyetlen ellenőrzési ponton keresztül, ugyanúgy, mint például az IBM Network SignON Coordinatorban, és ugyanígy egyetlen bejelentkezéssel elérheti az NT és az IntranetWare hálózati erőforrásokat. Ennek eredményeképpen a felhasználóknak nem kell aggódniuk a sok jelszó és felhasználói név miatt, valamint az NDS-en keresztül minden IntranetWare-szolgáltatáshoz hozzáférnek, többek között a fájlokhoz, nyomtatókhoz és alkalmazásokhoz.

Novell Workstation Manager

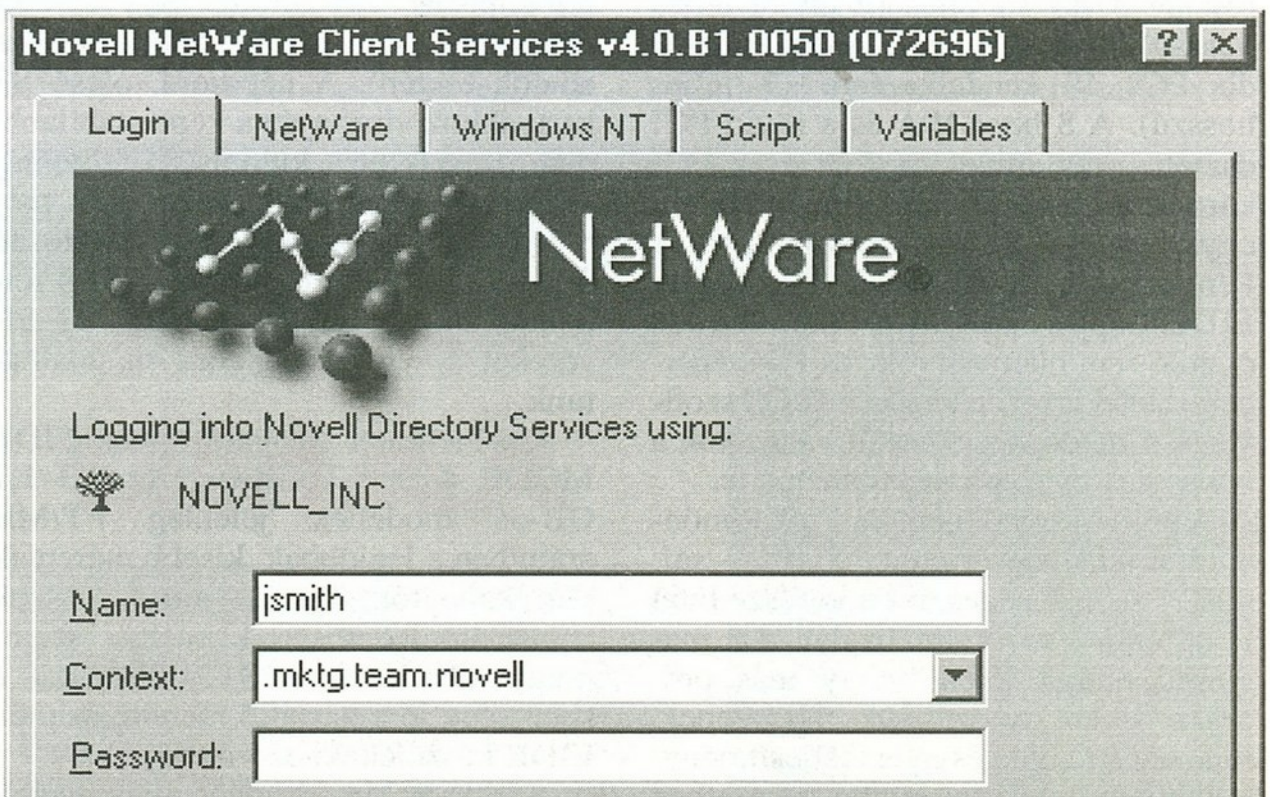
Legtöbbször az a probléma az NT és az (Intra)Netware integrációjával, hogy az NT klienseket NT tartományokon keresztül kell kezelni. A Novell kikerülő megoldása révén viszont elérhetővé válik az NT Server és a LAN Manager tartományok helyett használható, annál többet nyújtó NDS a Novell Workstation Managerrel, amely képes létrehozni dinamikusan felhasználókat, anélkül, hogy NT tartományokat kellene kezelni.

A GINA (IntranetWare Graphical Identification and Authentication) modul a Novell kliens alapkomponense Windows NT-n. A GINA az, amely végrehajtja az egyszeres bejelentkezést, bár valójában kétszer hajt végre ellenőrzést, egyszer a hálózaton, és egyszer a felületen.

Az NWAdmin segítségével a NetWare felhasználói jogok könnyedén kezelhetőek voltak egyetlen helyről, azonban NT alatt problémát jelenthetett az NT tartomány kikerülése, amikor NT munkaállomás jogokat kellett kezelni. Ha a tartományon keresztül folyt a munka, akkor az NDS-hez és a tartományi jogokhoz külön kellett hozzányúlni.

A dinamikus felhasználói jogok megkímélik az adminisztrátort a minden munkaállomásra kiterjedő egyenkénti létrehozástól. Ezenfelül, előfordulhat, hogy valaki publikus NT gépet szeretne létrehozni: ehhez nem kell több ezer potenciális felhasználót megadni, a dinamikus jogok létrejönnek, amikor a felhasználó belép az NDS-be.

Varga János



Nem luxus

Multimédiára kihegyezve

Az új hardvertermékekről szóló cikkek általában a vezető márkák szerelőszalagjairól éppen legördülő, és többnyire drága termékeket mutatják be. Kevés az olyan írás, amely a maga kategóriájában a legkeresettebb, azaz a többség számára könnyebben megfizethető kínálattal is foglalkozik. Ezért vettünk szemügyre most egy ilyen konfigurációt a Qwerty polcán.

A gép belsejében az alábbi dolgok lakoznak: Triton II VX alaplap, Pentium 133-as processzor, 32 MB RAM, ET6000-es videokártya, 1,7 GB merevlemez, 1,44-es lemezmeghajtó, 8x-os Sony CD-ROM és egy Genius SoundMaker 18 PnP hangkártya.

Az alaplap Gigabyte GA-586ATV típusú, 256 KB PB cache-sel felszerelve. A memóriának négy 36 bites SIMM bővítőhely áll rendelkezésére, és bár a VX chip-set támogatja az SDRAM-ot, ehhez szükséges DIMM slot nincs beépítve. Támogatja viszont az osztott feszültségű (dual vagy split voltage) processzorokat is (Pentium MXX, Cx 6x86L), így még egy generációt ki lehet bírni alaplapcsere nélkül. A kártyák részére 3 db 16 bites (2 ebből teljes hosszú), 1 db 8 bites ISA, valamint 4 db PCI áll rendelkezésre (3 teljes hosszú). A 8 bites ISA és a rövid PCI osztott, azaz mivel az ISA és a PCI kártyák hátlapja ellentétes irányban áll, egyszerre csak az egyik használható. A Pentium alaplapokon már megszokott EIDE, floppy, soros, illetve párhuzamos csatoló sem hiányzik róla, és PS/2 egérbemenettel is rendelkezik. A BIOS szoftveresen módosítható, az új változatok a www.giga-byte.com-ról tölthetők le.

A processzorról nem kell sokat mondani, a jelenlegi programokhoz — sajnos(?) — szükséges, mert jó része Intel Pentiumra van optimalizálva. Ez magyarul annyit jelent, hogy más processzorokon rosszabbul teljesítenek, mint azt a tényleges nyers teljesítménykülönbség indokolná, pedig sok esetben egy egyszerű újrafordítással a többi gépen is elfogadhatóan működne. A RAM típusa csak sima DRAM, pedig az EDO RAM gyorsabb valamivel (és általában olcsóbb is; valószínűleg nem volt raktáron). A memória mérete talán soknak tűnhet, de a mai árak mellett ez egyáltalán nem luxus, az új programok némelyike — különösen Win95/NT alatt — egyenesen követeli a 16 MB-nál több RAM-ot. Csak példaképpen: az MS Monster Truck Madness játéka induláskor figyelmeztet, ha a rendszerben 24 MB-nál kevesebb fizikai memória van!

A megjelenítésért egy ET6000-es chipen alapuló videokártya a felelős. Hasonló Hercules kártyáról már írtam a februári számban, a különbség az, hogy a driverek nem annyira kidolgozottak, és nem járnak hozzá szoftveres segédeszközök. Teljesítménye nagyon jó, a DOS alatti méréseknél minden felbontásban egyformán nagy átviteli sebességet produkált (85 MB/s 32 bites adatok esetén). A népszerű, olcsó S3 kártyákhoz viszonyítva képe stabilabb, nem csúszik el a különböző felbontásokban — igaz, többe is kerül. A memóriája 2,25 MB-os, azaz pontosan 2304 KB, ebbe még az 1024x768 felbontás 24 biten éppen belefér. Új drivereket a www.tseng.com-on találhatunk.

A winchester Seagate típusú EIDE MODE 4-es 1,7 GB-os. Az 1,3–1,7 GB-os modellek jelenleg FT/MB arányban a legjobbak, kisebb méretűek alig kaphatók, így itt nincs túl sok választási lehetőség kapacitás szempontjából. Ennek az ST31720A-nak a sebessége jó a hasonló méretű, szintén EIDE készülékekhez viszonyítva.

A CD-ROM szintén alapkövetelmény, mert — egy-két kivételtől eltekintve — minden új program CD-s, hiszen sorozatgyártásban a CD olcsóbb, mint a floppy, és kevésbé érzékeny. Szerintem egy 4x-es CD is megfelelő az átlagos használatra, de jelenleg 8x-osnál lassabb meghajtó nem kapható, a Sonynak ezek között pedig jó a híre. A sebessége 1198 KB/s, átlagos elérési ideje 145 ms. A Genius hangkártyája az ESS1868 chipen alapul, ez a chip található szinte minden olcsóbb hangkártyán jelenleg. DOS alatt SB Pro (8 bit, sztereó) kompatibilis, Windows alatt 16 bites sztereó 44 kHz-es módban működik. Drivere van Windows NT-hez is, ami nem megszokott. Hangja nem a legtisztább, a szünetekben különösen hallani a kártya zúgását, ezért kényes ízlésűeknek nem ajánlott.

A monitort szándékosan hagytam ki a felsorolásból, mert minőségben és — sajnos — árban is egy magasabb kategóriát képvisel. A Samsung 15 colos SyncMaster 15GLi modellje egy digitálisan, képernyőn megjelenő menük által beállítható, igazi kis munkamonitor. (Munkamonitoron azt értem, hogy akár egész nap előtte lehet ülni, anélkül, hogy fejfájást kapnék.) A beállítások — a hagyományosan tekerős fényerőn és kontraszton kívül — nyomógombos felületről történnek. Az állítható tulajdonságok a képméret és elhelyezkedés, a hordó- és trapéztorzítás, valamint a színhőmérséklet (ez két előre rögzített paraméterből választható). Lehetőség van a gyári adatok visszahívására a recall funkcióval, és van degauss gomb, amely az esetleges mágneses torzulásokat hivatott megszüntetni.

A teljes konfiguráció ára március végén, monitor és áfa nélkül 144 ezer Ft volt. Ez elsőre soknak tűnt, de egy kis utánaszámolással könnyen belátható, hogy ennél sokkal olcsóbban otthon sem lehetne összerakni, ráadásul akkor a garancia csak 1 év, viszont a Qwerty másfelet ad. (Az összeszerelő cégek közül igen kevés kínál az alkatrész garanciájánál hosszabb teljes körű jótállást.) A monitor ára már riasztóbb, 76 ezer Ft szintén áfa nélkül, ezért a legolcsóbb 17 colost is meg lehet kapni, ami azért elgondolkoztató — én inkább egy olcsóbbat vennék.

Ez a gép a jelenlegi multimédiás igényeknek kíván megfelelni, amihez minden tartozéka megvan. Aki nem szeret vagy nem mer saját maga bővítgetni, annak egy ilyen összeállítás ideális lehet. Ha valaki szeretne egy kicsit spórolni egy hasonló konfigurációból, akkor talán a RAM feléről — ha munkához kell, akkor a hangkártyáról is — lemondhat, esetleg a videokártyát kicserélheti egy olcsóbbra. Mást nem érdemes, mert az már a használhatóság rovására menne.

Bánó György

Umax síkágyas szkennerek

A „képüzem" bejárata

A Umax a kisebb multimédia-kiegészítők gyártása után a DTP-eszközök piacán előbb professzionális szkennerekkel, majd Macintosh-hasonmásokkal jelentkezett. Ez utóbbiak gyártásának jogát nem is olyan könnyű megkapni: igen szigorú feltételekhez köti az Apple. Cikkünk az előbbiekre koncentrál.

A most nagyító alá vett két síkágyas szkenner a SuperVISTA S-12, valamint a legújabb Astra sorozat — amely a SuperVISTA családot fogja felváltani — 600S típusa. Ez utóbbinak a 600P néven megkülönböztetett párhuzamos portos változata paramétereiben megegyezik a 600S típussal.

Az S-12 a jelenlegi Umax szkennerválaszték egyik félprofi típusa, optikai felbontása 600x1200, színmélysége 24 bit, és tartalmaz egy analóg gammakorrekciós áramkört, amelynek a segítségével — az ismertető szerint — a képek minősége 33 bitesnek felel meg; a legnagyobb beolvasható laptípus a szabványos A4-es. Az ilyen kategóriában már megszokott diafeltét és lapadagoló is kapható hozzá (a készülék árának feléért/kétharmadáért).

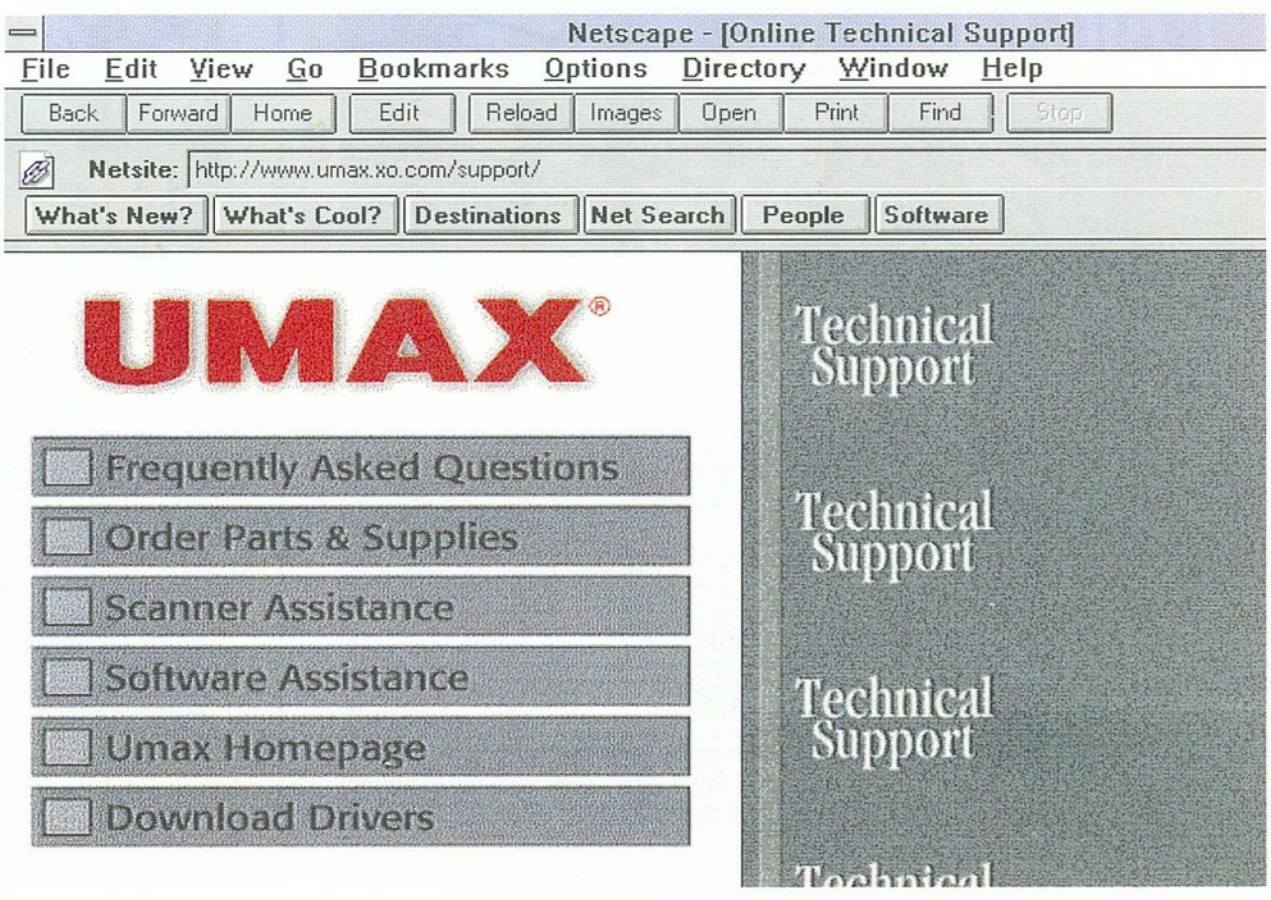
Az S-12 dobozában a szkenneren kívül a hálózati és SCSI kábel, egy CD, egy gyors telepítési útmutató, valamint az igen vékony kezelési utasítás található. A kezelésre vonatkozóan nincsenek különleges követelmények, csak a síkágyas szkennerek szállításánál használt olvasófej-rögzítő csavart kell kilazítani, mielőtt a hálózatba kapcsoljuk. SCSI kártya nem volt tartozéka a kapott csomagnak; akinek nincs, meg kell vásárolnia egyet. A CD-n a PC- és a Mac-szoftver mellett a programok komplett leírása kapott helyet.

A telepítés egyszerűen zajlott — úgy látszik, lassan tényleg nem kell szakembernek lenni egy ilyen művelethez —, viszont kérdezés nélkül beleírt az autoexec.bat-ba, bár csak a Path-hoz fűzte hozzá a saját könyvtárát. A programok dokumentációja Adobe Acrobat (.pdf) formátumú, és amennyiben nincs ehhez szükséges olvasó a rendszerben, felajánlja annak a telepítését is. (Igaz, hogy környezetkímélő dolog nem papíron adni a leírásokat, de azt hiszem, nem vagyok egyedül, aki inkább a nyomtatott használati utasításokban szeret keresgélni, a digitális változatot igazán csak a helyzetérzékeny helpek esetében tudom becsülni.)

A mellékelt szoftvereket a Presto! Pagemanager keretprogram fogja össze. A Pagemanagerből közvetlenül faxolhatunk, e-mailt küldhetünk, illetve egy egyszerű — angol — szövegfelismerőt használhatunk, míg a képfeldolgozó PageImage és az űrlapkitöltő PageType külső programokat egy ikonsorból indíthatjuk, drag-and-drop módszerrel. Az ikonsor tetszőleges alkalmazással bővíthető. A szkennelés csak a Pagemanager programból indítható (a mellékelt programok közül), ami a Twain-kompatibilis VistaScan modulon keresztül történik. Ennek során beállíthatjuk a szokásos paramétereket: a terület méretét, a kívánt felbontást, színmélységet — sőt kontrasztot és fényerőt is —, valamint a dokumentum típusát (papírkép, negatív vagy dia).

Természetesen előzetes (preview) lehetőség is van a könnyebb munka érdekében; sőt ha kilépünk a programból, vagy akár a Windowsból, a legközelebbi alkalommal is az utoljára beolvasott kép előzetese lesz látható szkenneléskor. A színmélységről érdemes megjegyezni, hogy true color esetén nemcsak RGB formátumban képes beolvasni, hanem CYMK-ban is, amivel nyomdai előkészítő munkáknál műveletek takaríthatók meg. Lehetőség van még a szkennelési színek finomítására is — automatikusan vagy kézileg, egy görbe megadásával; ez azért lehet szükséges, hogy a képernyőn vagy a nyomtatón megjelenő színek megfeleljenek az eredetinek. Ha már beolvastuk a képet, akkor a kívánt program ikonjára kattintva behívhatjuk az alkalmazásokat. Azt, hogy a program milyen képformátummal tud dolgozni, minden programra külön lehet megadni, például a PageImage alapértelmezett formátuma a JPEG.

A képfeldolgozó program lehetőségei nem túl nagyok, de az alapműveleteket ezzel is el lehet végezni. Akinek jobb program kell, az a szkenner mellé kb. fele áron beszerezheti a Photoshop különböző változatait. Aki a teljes vál-



tozatot megveszi, az később élhet az upgrade lehetőségével is, és ez nem túl gyakori az ilyen árú szoftverek akciós vásárlásánál.

Az Astra 600S az új sorozat első darabja, ezt hamarosan követi majd a 600P és az 1200S — az S-12 megfelelője. Ez a típus kapható Mac és PC csomagban is, lényeges különbség, hogy a PC-s változat tartalmazza az SCSI vezérlőt is. Ez a szkenner viszont már nemcsak A4-esig olvas, hanem a nálunk „kifutó A4-esnek" ismert ún. legal méretben is. A mellékelt programok a előzőekben már ismertetett Presto! Pagemanager, valamint a Umax Copy Utility — ennek segítségével fénymásolóként használhatjuk a szkennerünket, természetesen ha nyomtató is csatlakozik a géphez. A telepítés azonos a S-12-esével, hiszen a főbb programok is ugyanazok. A Copy Utilityt nem igazán tartom fontosnak, mert az igazi másolók üzemeltetése olcsóbb, és már rengeteg iroda fel van szerelve ezekkel. Esetleg színes másolatnál lehet érdekes, viszont ahhoz jó minőségű színes nyomtató kellene, ami ritka és drága.

A beolvasott képek minősége véleményem szerint mindkét esetben inkább az eredeti minőségétől függött, semmint a szkennerek képességétől. A Vistascan modul megfelelő beállításaival elérhető, hogy a kép színét, kontrasztját és fényerejét még a beolvasás előtt finomítva, szinte publikálásra kész anyagot kapjon az ember.

A többi program inkább csak addig hasznos, amíg az ember nem szánja rá magát egy komolyabb editor beszerzésére.

Az S-12 modell igen megnyerő paraméterekkel rendelkezik, de szerintem az ára — ha nem vesszük figyelembe a szoftverkedvezményeket — kissé magas az egyszerűbb irodai típusokhoz viszonyítva. Ott lehet érdemes a megvásárlása, ahol komoly munkához szükséges szoftver még nem áll rendelkezésre, de a szkennerrel együtt kívánják beszerezni, ezzel a többletárat a kedvezményes program ellensúlyozza. A 600S megfelelő lehet ott, ahol nem kell 600 dpi-nél nagyobb felbontással dolgozni (weblapokon és egyéb digitális formában megjelenő médiákban általában elegendő a 300 dpi is). Ennek ára már azonos szinten van a kisebb lézernyomtatókéval, így ahol azt megengedhetik maguknak, ott már nem luxus egy ilyen készülék, persze csak ha tényleg szükség van az alkalmazására.

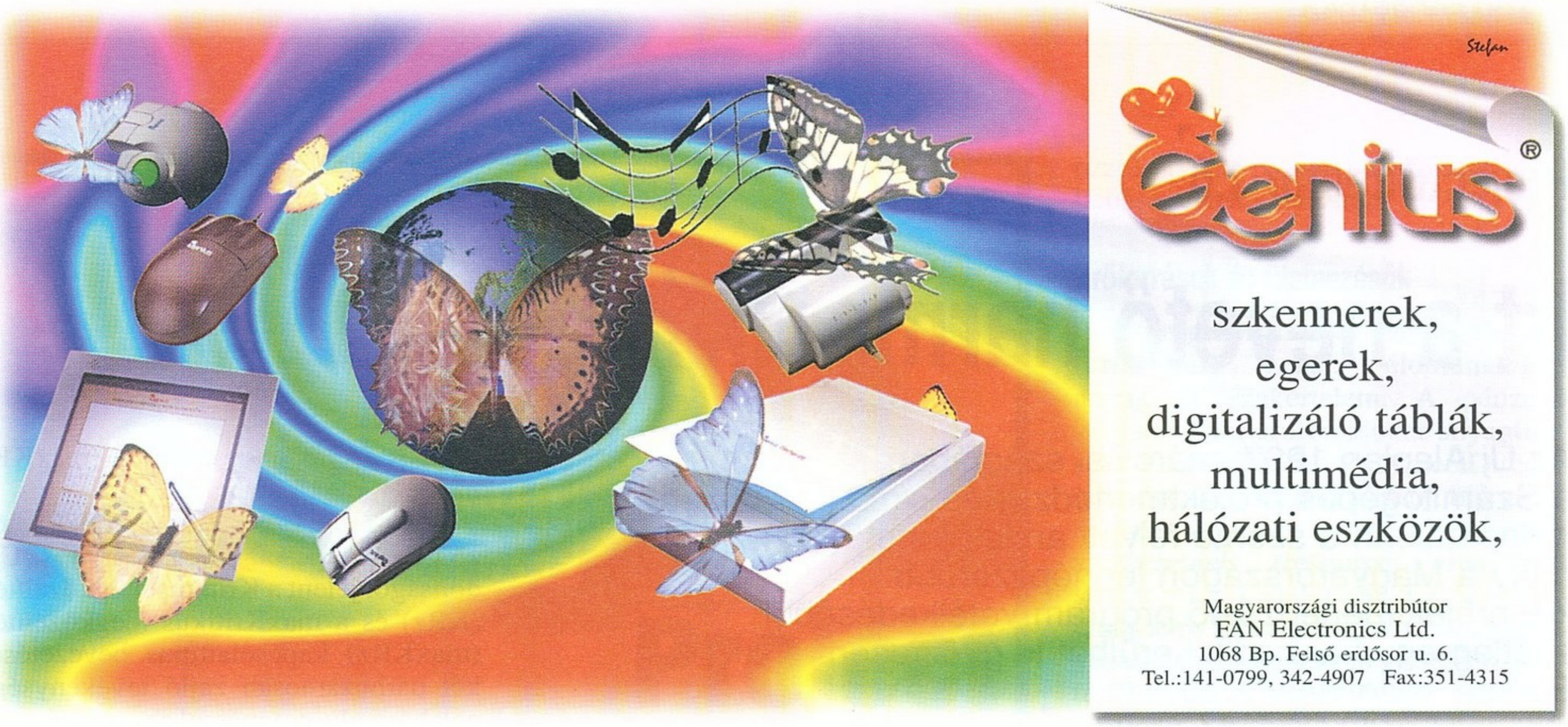
Egyéb információkat, képeket — különböző pályázatokból a nyerteseket! — és a termékek drivereit találhatjuk meg a cég weboldaláról elindulva (www.umax.com).

Bánó György

Technikai adatok:

Típus	Astra 600S/P	SuperVISTA S-12
Optikai felbontás (dpi):	300x600	600x1200
Szoftveres felbontás (dpi):	4800	9600
Színek (bit):	30	24
Leolvasható max. méret (mm):	216x356	216x297
Ár (Ft):	79.900/74.900	181.500

INFORMÁCIÓKÉRÉS: 0515 ▼



INFORMÁCIÓKÉRÉS: 0535 ▼



INFORMÁCIÓKÉRÉS: 0520 ▼



ProjectDirector 1.0

Itt a nevető harmadik?

Az Új Alaplap 1997. márciusi számában, a Számítógépes projektmenedzselés című írás utolsó mondataiban a szerző felveti annak lehetőségét, hogy a Magyarországon legnépszerűbb két projektmenedzselő program vetélkedéséből esetleg egy harmadik kerülhet ki győztesen. És íme, megjelent a piacon a magyar alkotógárda által kifejlesztett ProjectDirector, amely méltán veszi fel a versenyt a Microsoft Project és CA-SuperProject programcsomagokkal.

A projektek tervezése, ütemezése, a megvalósítás irányítása bonyolult feladat. A projektcél elérését, valamint annak megvalósításához a rendelkezésre álló költségkeretet, az erőforrásokat és a határidőket összhangba kell hozni. Ha az operációkutatási módszerek oldaláról vizsgáljuk a kérdést, azt találjuk, hogy induláskor rendszerint ellentmondásban van a feladat és a feltételrendszer. Ennek megszüntetésére és az optimális ütemezés kialakítására nincs egzakt matematikai algoritmus. Ilyenkor kerül sor az ún. heurisztikus módszerek alkalmazására, amelyek egyrészt segítenek az ellentmondások feltárásában és megszüntetésében, másrészt egy „jó" megoldást eredményeznek. Mivel a feltételrendszer és esetleg a projektcél módosítása nem történhet automatikusan, a projekt végleges terve egy iterációs folyamat eredményeként alakul ki, amelyben a tervező lépésről lépésre finomít az adatokon, az adatokból felépített hálóterves modellen, és a program paramétereinek beállításán.

Ehhez a folyamathoz nyújt számítástechnikai hátteret a ProjectDirector is. A programrendszer — hasonlóan a többi projektmenedzsment szoftverhez — támogatja a modell összeállítását, a tevékenységek ütemezését, az erőforrások elosztását, a költségek alakulásának vizsgálatát, a tervvariánsok kidolgozását. Lehetőséget nyújt a megvalósítás közben felmerült módosítások követésére, valamint a tervezett és a tényállapotnak megfelelő ütemtervek összehasonlítására. A kinyomtatott Gantt-diagramok, erőforrás-terhelési görbék, táblázatok segítik a projekt irányítását és a projekt-team tagjai közötti együttműködést, információcserét.

A következőkben áttekintjük a program működésének főbb területeit, de elsősorban az újdonságokat, érdekességeket emeljük ki.

A modell

A hálóterves modellben a projektet részfeladatokra bontjuk, és meghatározzuk az egyes részek kapcsolatát a többivel. A tevékenységek és kapcsolataik ábrázolására a program a napjainkra leginkább elterjedt MPM technikát használja.

A ProjectDirector egyik fő erőssége az ún. maximális kapcsolatok ábrázolási lehetősége, amellyel a hasonló funkciójú programoknál nem találkozunk. Ennek lényegét egy egyszerű példával kíséreljük meg érzékeltetni.

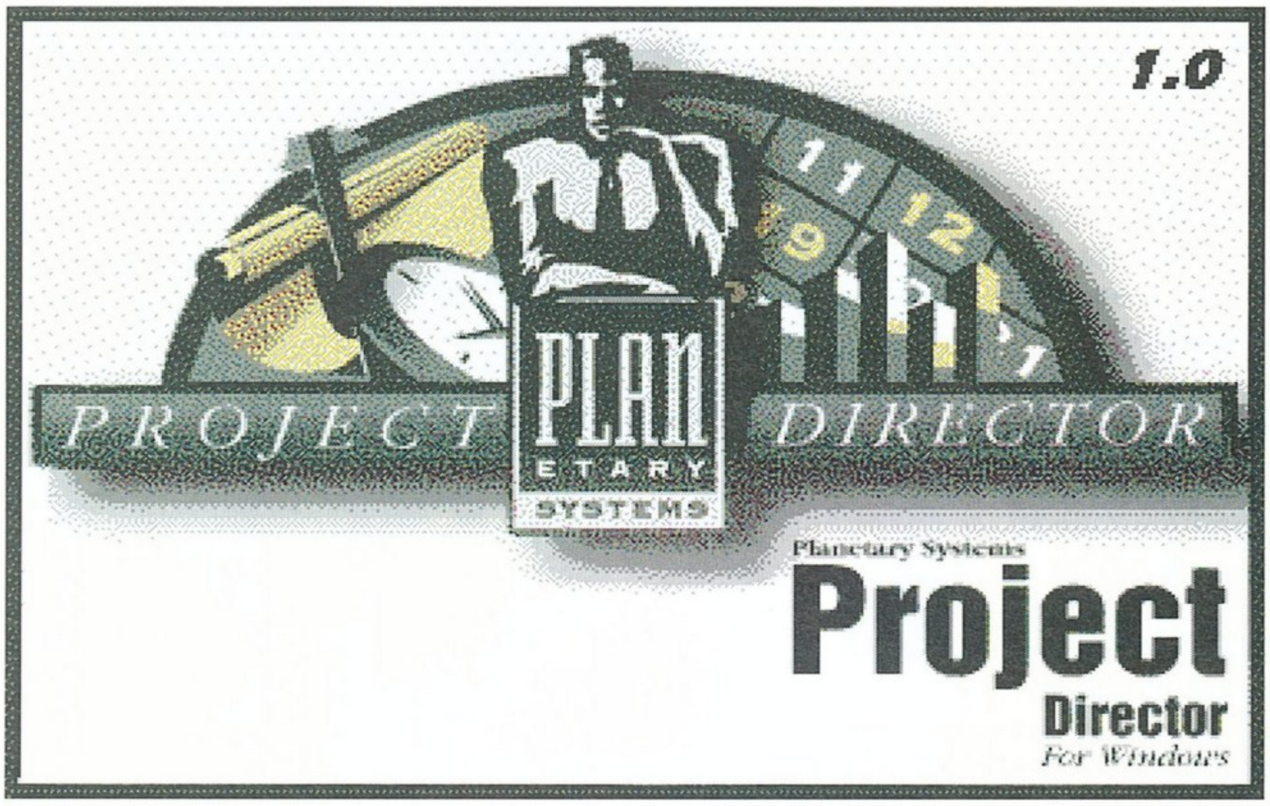
A hagyományos MPM kapcsolatok két tevékenység között a megelőzési viszony meghatározása mellett a tevékenységek minimális időbeli távolságát adják meg. Így például az alapgödörkiemelés és a dúcolás között megadott BK2 kapcsolat előírja, hogy a kiemelés befejezése után legkevesebb 2 napnak kell eltelnie a dúcolás megkezdéséig. A maxBK3 kapcsolattal viszont azt is megadhatjuk, hogy a megelőző tevékenység vége és a követő tevékenység kezdete között legfeljebb 3 nap telhet el. Ezzel gondoskodunk arról, hogy a kiásott munkagödör ne omoljon be a dúcolás előtt. A maximális értékeket természetesen mind a négyféle kapcsolattípus esetén használhatjuk (maxKK, maxKB, maxBK, maxBB).

Lehetőség van két tevékenység között több kapcsolat megadására is. Újdonságot jelent a Kritikus megközelítés (KRz) és a maxKritikus megközelítés (maxKRz) kapcsolattípus bevezetése; két párhuzamosan zajló tevékenységnek bármely, egymással azonos készenléti foka közötti időbeli távolság minimumát, illetve maximumát írják elő. Az új kapcsolattípusok bevezetése miatt a klasszikus időelemzési algoritmust is módosított formában alkalmazza a program, folyamatosan segítve a felhasználót az esetleges tervezési hibák feltárásában.

A tevékenységazonosítók

Már a tervezés első fázisában gondolnunk kell arra, hogy az ütemezés eredményeinek jól áttekinthető formában, megfelelően kiválogatva, csoportosítva kell megjelenniük. Ehhez információk megadása szükséges. A tevékenységadatok felvitelekor részekre bontott (szegmentált) tevékenységazonosítókat adhatunk meg. Például egy több épületből álló kis lakótelep esetén az első szegmens az épület azonosítóját tartalmazza, a második annak a szintnek a számát, ahol a tevékenység zajlik, a harmadik a megvalósító szervezetet, a negyedik a munkafolyamatot jelöli. Az így kialakított azonosító-rendszerből a tevékenység szöveges megnevezése nélkül is könnyen leolvashatók a legfontosabb információk.

Az így felépített kód alapvető célja a szelektív output elkészítésének lehetőségét biztosítani. A projekt operatív irányítóit nem terheljük felesleges, őket nem érintő adatokkal, a vezetők pedig területüknek és szintjüknek megfelelő összevont ütemterveket, kimutatásokat kaphatnak. A projektek tevékenységeinek tervezésekor gyakran használt top-down tervezési eljárást támogatja a program. Az így kialakult, összevont és alárendelt tevékenységekből álló hierarchikus rendszert *munkalebontási szerkezetnek* (work breakdown structure, WBS) nevezzük. Ezzel a szerkezettel összefüggő kód a WBS-kód. A program a szegmentált tevékenységa-



zonosítók felhasználásával, a munkalebontási szerkezet alapján automatikusan határozza meg a WBS-kódot.

A tevékenységek típusai

A ProjectDirectorban négy tevékenységtípust használhatunk. A normál tevékenységen kívül megadhatunk ún. *befüggesztett tevékenységet* (hammock activity), amelynek időtartamát a program számolja ki végpontjai időbeli elhelyezkedésének függvényében. Ez jól használható az időtartamában előre nem ismert tevékenységek (például területfoglalás, víztelenítés stb.) erőforrásigényének, költségének és rezsijének a meghatározásához. A *kezdés mérföldkő* (start milestone) és a *befejezés mérföldkő* (finish milestone) nulla időtartamú tevékenységek, amelyek egy fontosabb projektfázis kezdetét, illetve befejezését jelzik.

A tevékenységek időtartama lehet rögzített (fixed duration) és erőforrás-vezérelt (resource driven duration). Az első esetben a tevékenységidő nem változik akkor, ha módosítjuk a tevékenységhez rendelt erőforrásokat. A második esetben (az elvégzéshez szükséges munkát állandónak tekintve) a program a megadott erőforrásigények változtatásakor megfelelően módosítja a tevékenység végrehajtási időtartamát.

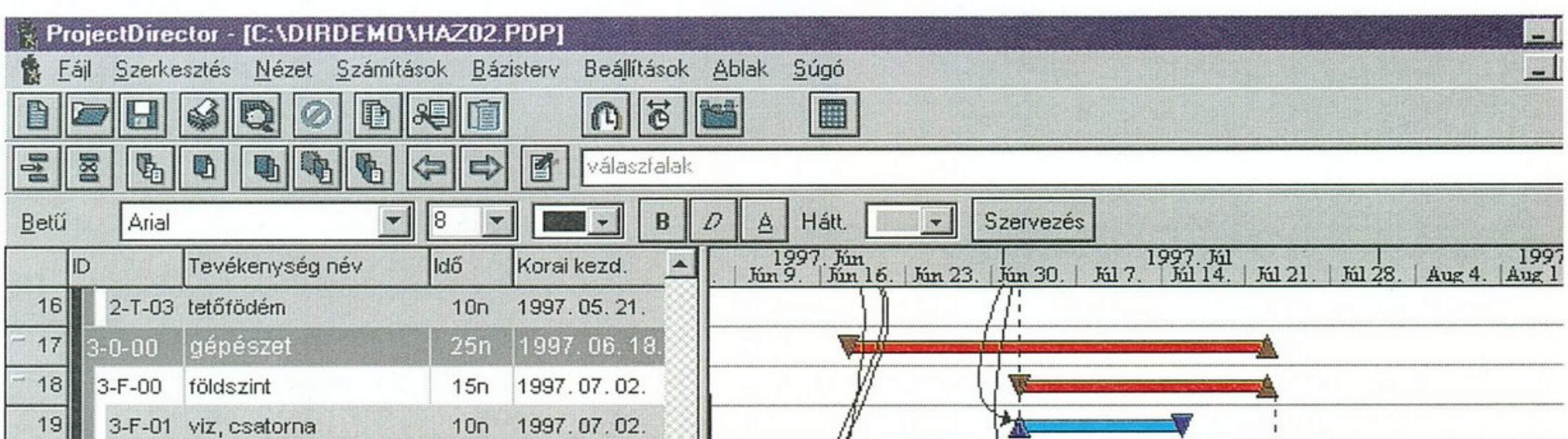
Különlegességnek számít, hogy az *együtt dolgozás* paraméterrel előírhatjuk a tevékenységen dolgozó erőforrások egyidejű felhasználását. Ez azt jelenti, hogy a tevékenységet csak olyan időpontra ütemezi a program, amikor a tevékenységhez rendelt összes erőforrás rendelkezésre áll. A *kezdd másnap* paraméterrel azt állíthatjuk be, hogy a tevékenység csak a munkanap kezdetén indul, függetlenül attól, hogy a megelőző tevékenység az előző munkanapon mely órában fejeződött be. Szintén különleges előnyt jelent a *megszakítható tevékenységek* megadásának lehetősége. Ennek kiemelt jelentősége van az átfedések és a nem azonos ütemű tevékenységek szinkronizálásában. Ez a lehetőség a valós helyzet pontosabb modellezésével a terv átfutási idejének, ezzel együtt a projekt költségének csökkenését eredményezheti.

Erőforrások és ütemezésük

A programrendszer háromféle erőforrástípust kezel: élőmunka/gép, anyag és szakértelem. A pénzeszközöket mint erőforrásokat anyagtípusúként vihetjük fel. A ProjectDirector lehetővé teszi az erőforrások hierarchiájának, azaz az *erőforrás-lebontási szerkezetnek* (resource breakdown structure) a kialakítását. Ezt a hierarchiát létre lehet hozni akár a cég szervezeti felépítésének megfelelően is. A tevékenységekhez összegző, vagyis az erőforrás-hierarchia magasabb szintjén lévő erőforrásokat is hozzá lehet rendelni. Egy erőforrással kapcsolatban többek között a következő adatokat adhatjuk meg: név, típus, fajlagos óraköltség, egyszeri felhasználási költség, túlóraköltség, szakértelem (skill).

Ez utóbbit néhány mondatban bemutatjuk, mivel más szoftverekben ezzel a megoldással nem találkozhatunk. Megadhatjuk, hogy milyen szakértelemmel rendelkeznek a megvalósító szervezetnél. A szakértelmet később élőmunka/gép típusú erőforrásokhoz kapcsolhatjuk, egy erőforráshoz akár többet is. (Például Karádi Pál hegesztő és lakatos.) A tevékenységek erőforrás-szükségleteinek meghatározásánál elegendő megadni a szakértelmet, amennyiben a feladat elvégzéséhez szükséges munkaerőt nem ismerjük pontosan.

A program erőforrás-korlátos és időkorlátos ütemezést tud készíteni. Kiválaszthatjuk azokat az erőforrásokat, amelyekre a tervezést el akarjuk végezni.



A naptárak

A naptárak a munkaidő-tervezés nélkülözhetetlen eszközei. A ProjectDirectorban tetszőleges számú naptárt definiálhatunk a tevékenységekhez és az erőforrásokhoz. A naptárak hierarchiába rendezhetők, az új naptár könnyen származtatható egy meglévőből, akár egy munkanapot ünnepnapra változtatva. A különböző napok, periódusok megadásához 26-féle definíciót használhatunk, például minden csütörtök, vagy minden év hetedik hónapjának első hete.

Tevékenységfeltételek

Vannak olyan külső körülmények, amelyeket a modellben nem tudunk ábrázolni. Ezek figyelembevételére leggyakrabban azt a megoldást választjuk, hogy a tevékenység kezdetét, befejezését valamilyen dátumhoz kötjük. A program elsődleges feltételekként kezeli a szokásos nyolc korlátot (legkorábbi kezdés; legkésőbbi kezdés; kezdés ekkor; befejezés ekkor; kezdd később, mint...; kezdd korábban, mint...; fejezd be később, mint...; fejezd be korábban, mint...).

Szintén újdonság, hogy másodlagos feltételek is megadhatók a modellben. Itt is az előbb említett korlátok alkalmazhatók, de hatásukra a program csak a legkorábbi és a legkésőbbi időpontok között mozgatja a megadott tevékenységeket.

Képernyők és beszámolók

A projekt adatai és az ütemezés eredményei a szokásos Gantt-diagramon és táblázatokon kívül megtekinthetők naptári, PERT ábra és WBS ábra formájában is. Az összes képernyőkép és tábla testre szabható és kinyomtatható. A betűk típusa, stílusa, mérete, színe, a diagramok formája, mintája beállítható. Igazi különlegesség az íves tevékenységkapcsolatok megadásának lehetősége.

A felhasználó által akár egyenként állítható Bézier-görbékkel kiküszöbölhetők az egymást fedő kapcsolati vonalak. A létrehozott beállítások átvihetők más projektekbe is, ami rendkívül gyors munkát tesz lehetővé.

A programcsomagban jól használható kézikönyvek is vannak, amelyek a program kezelésének leírása mellett az elméleti tudnivalókat is összefoglalják. A programrendszerbeli szakmai újdonságok a Hajdu Miklós által írt Network Scheduling Techniques for Construction Project Management című könyvben részletesen megtalálhatók. Ezt a mintegy 350 oldalas angol nyelvű szakkönyvet a Kluwer Academic Publishers kiadó ez év elején jelentette meg.

A budapesti Planetary Systems Kft (www.plansys.hu) programozói gárdája korszerű módszerekkel és fejlesztőeszközökkel készítette el a programrendszert. Reméljük, hogy az 1.0-s változatot követi a 2.0, 3.0 ..., amelyekben az itt most nem részletezett kisebb hiányosságok megszűnnek. A magyar és angol nyelven elkészített programrendszer már Párizsban (IPMA '96) és Hannoverben (CeBIT '97) is sikeresen bemutatkozott. A hazai alkalmazások közül kiemelkedik az új Nemzeti Színház felépítésének tervpályázatához készült ütemterv.

Tátrai Tibor
tatrai@mail.matav.hu



INFORMÁCIÓKÉRÉS: 0503 ▼

PROFON

INFORMATIKAI Kft.

AMI MINDIG MINDENT ÖSSZEKÖT

KOMPLETT SZÁMÍTÁSTECHNIKAI ÉS HÍRKÖZLŐ HÁLÓZATOK TERVEZÉSE ÉS KIVITELEZÉSE

GYENGEÁRAM

- ADATHÁLÓZATOK
  - UTP/FTP strukturált rendszerek
  - Ethernet
  - Twinax
  - Type 1. Cabling System
  - Optikai
- HÍRKÖZLŐ HÁLÓZATOK

ERŐSÁRAM

- SZÁMÍTÁSTECHNIKAI RENDSZEREK FÜGGETLEN ENERGIAELLÁTÁSA
- SZÜNETMENTES TÁPEGYSÉGEK TELEPÍTÉSE
- FELÜLVIZSGÁLAT, BEMÉRÉS
- TÚLFESZÜLTSÉG ELLENI VÉDELEM

ETHERNET és TOKEN RING HÁLÓZATI ESZKÖZÖK

- ROUTEREK
- HUBOK
- SWITCHING HUBOK
- REPEATEREK
- SZÓRT SPEKTRUMÚ HÁLÓZATI ESZKÖZÖK

HÁLÓZATÉPÍTŐ ANYAGOK

Rövid kivitelezési határidő, hároméves garancia!

Kérje árlistánkat a FAXBANK-ból: 180-8611/1901

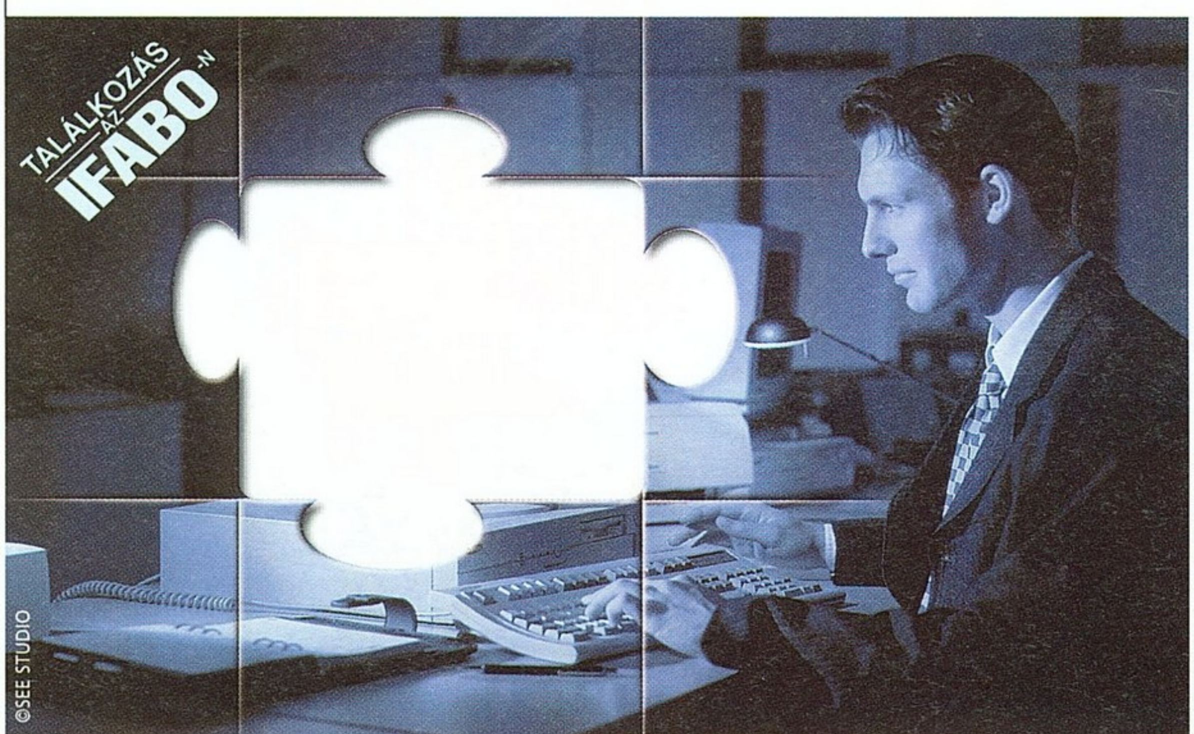
1138 Budapest Cserhalom út 4.
Telefon: 270-6227, 270-6235
Telefon/Fax: 270-50-93
E-mail: profon@mail.matav.hu

IFABO B pav. 10.

MASTER'S Grafika

INFORMÁCIÓKÉRÉS: 0547 ▼

ÚJ GENERÁCIÓ - A G SOROZAT



A **G SOROZAT** 14-féle monitor modellje már az ezredvég technológiai követelményeire van felkészítve. Az **USB** opció biztosítja a monitorok hosszú távú használatát azáltal, hogy a változatos és gyorsan fejlődő perifériákkal összhangot teremt. A Samsung SyncMaster™ 500s, 700s, 500b, 700b monitorok az otthoni munkavégzés a kis- és középvállalkozások ideális választása.

A **G SOROZAT** minden tagja multimédia-változatban is kapható. A sorozat modelljei az ergonómiai követelményeknek megfelelően **TCO** kivitelben is választhatóak. A monitorokra **három év teljes körű garanciát** biztosítunk.

USB UNIVERSAL SERIAL BUS · Display Director

...ha a gazdaságosság a szempont



S - Small/Home Office B - Business M- Multimedia

SAMSUNG ELECTRONICS

SAMSUNG BEMUTATÓ TEREM:
1081 Budapest, József krt. 13.
Telefon: 117-4776

INFORMÁCIÓKÉRÉS: 0554 ▼





INFORMÁCIÓKÉRÉS: 0514 ◄



INFORMÁCIÓKÉRÉS: 0526 ◄

C++ Builder

A Borland bemutatja...

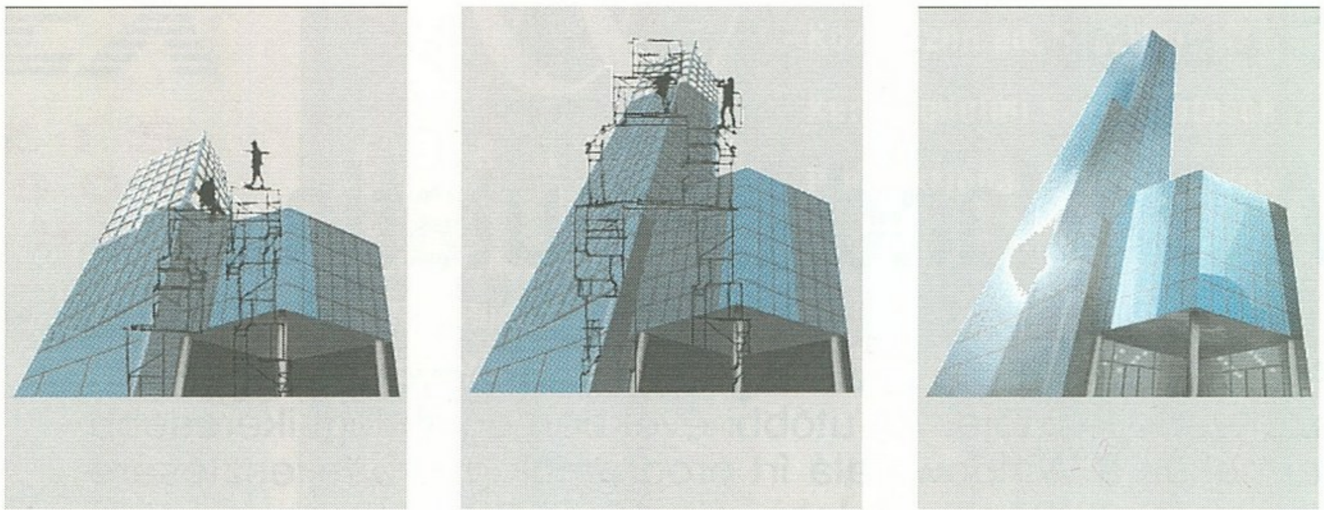
A Borland még ma is a világ egyik legnagyobb független szoftverfejlesztője. Az utóbbi években egyik legsikeresebb termékük a Windows alá írt programok gyors fejlesztésére alkalmas eszköz, a Delphi volt, 16 és 32 biten egyaránt. Ennek kiadásával új programfejlesztési koncepció jelent meg a Pascal-fordítók, Pascal-alapú fejlesztőkörnyezetek körében, az objektumorientált vizuális alkalmazásfejlesztés, melyhez új „programnyelv", az Object Pascal kialakítása is társult. A Delphi 32 bites verzióját követően (Borland Delphi 2.0) a C++ nyelven fejlesztők számára most megjelent a Delphi képességeit megvalósító, de a C nyelvet támogató új fejlesztőeszköz.

Az új termék budapesti bemutatójára március közepén került sor. A Borland C++ Builder-alapú termékcsalád (fejlesztőkörnyezet) a 32 bites Windows-verziók (Win95, Windows NT) számára készült. A korábbi verziókat nem támogatja. A telepített Borland C++ Builder mint fejlesztőkörnyezet a Windows alatt a Delphiben megszokott képet mutatja, s mint a bemutató során kiderült, a programfejlesztés alaplépései is jobbára megegyeznek az említett Object Pascal környezetben végzett fejlesztéssel. A vizuális elemeket ugyanúgy „fogd és vidd" technikával lehet elhelyezni a projekt alapját képező FORM-on. Eközben — mint a Delphiben — komponenslistáról lehet kiválasztani a kívánt elemeket, és ezek tulajdonságait is a Delphiben megszokott „Object Inspector" paneljén kísérhetjük figyelemmel, illetve ott folyamatosan módosíthatjuk is azokat. A Borland C++ Builder ezzel igen alkalmas a gyors alkalmazásfejlesztésre.

Természetesen a Borland C++ Builder is támogatja a szövegformátumú forráskód megírását, módosítását. Ezzel a gyárilag nem meglévő funkciók megírhatók, a meglévők megfelelően módosíthatók. A kész program e forrás fordításával áll elő, mint valódi .EXE fájl. A fordítás gyors. Ennek oka, hogy bár a projekt első fordítása lassabb, de ezt követően már csak az eltérések kerülnek fordításra, ami a későbbiekben jelentősen mérsékli a fejlesztési, kipróbálási időt.

A Borland gondolt azokra is, akiknek esetleg már jelentős készletük gyűlt

össze a Delphi alá fejlesztett, és abban is a programfejlesztés egyik alapját képező komponensekből. Először kicsit szokatlannak hangzó módon a jelen új fejlesztőkörnyezet támogatja a Delphi komponenseinek használatát. Ezzel a megoldással átjárást biztosít a két fejlesztőkörnyezet között, lehetővé téve akár vegyes „nyelvű" fejlesztőcsoportok eredményes munkáját. A Delphi kész, forrásnyelven rendelkezésre álló elemeinek módosítására azonban a Builder környezetében nincs lehetőség. Ennek áthidalására viszont a Delphiben írt részek közvetlenül átemelhetők a C++ projektbe. Ekkor már lehetőség van a módosításra is. Természetesen a Pascal forrás ebben az esetben is érintetlenül marad.



Az alkotók számoltak az adatbázis-alkalmazások gyors fejlesztésének igényével is. A termék támogatja az SQL-t, és Query nevű komponensének felhasználásával egyszerűen elvégezhetjük ezen alkalmazások kialakítását, akár különböző gépeken található, különböző formátumú adatbázisokból is összegyűjtve az adatokat. Az adatbázis-alkalmazás felépítésénél egyszerűbb esetben manuális kódírás nélkül is megvalósítható egy teljes alkalmazás kialakítása. A mezőket néhány előkészítő lépés után szintén „fogd és vidd" módszerrel helyezhetjük el az alkalmazás ablakába, és az adatbázisban való navigáláshoz is önálló eszközünk van. Természetesen teljes táblázatok elhelyezésére is lehetőség van, amelyekhez a táblázatkezelőkhöz hasonló vezérlőelemek állnak rendelkezésre.

Hálózati alkalmazások fejlesztésekor MS Explorer, illetve az OLE megoldásait egyaránt alkalmazhatjuk, és lehetőség van különböző aktív vezérlők alkalmazására. Amennyiben ezek nem a Windows alaprendszeréhez tartozó elemek, „illik" mellékelni az alkalmazáshoz. Komplett támogatást ad a HTML formátum kezelésére. Ahogy a budapesti bemutatón is látható volt, az e célra rendelkezésre álló komponens felhasználásával könnyen alakítható ki egy egyszerűbb web-böngésző is.

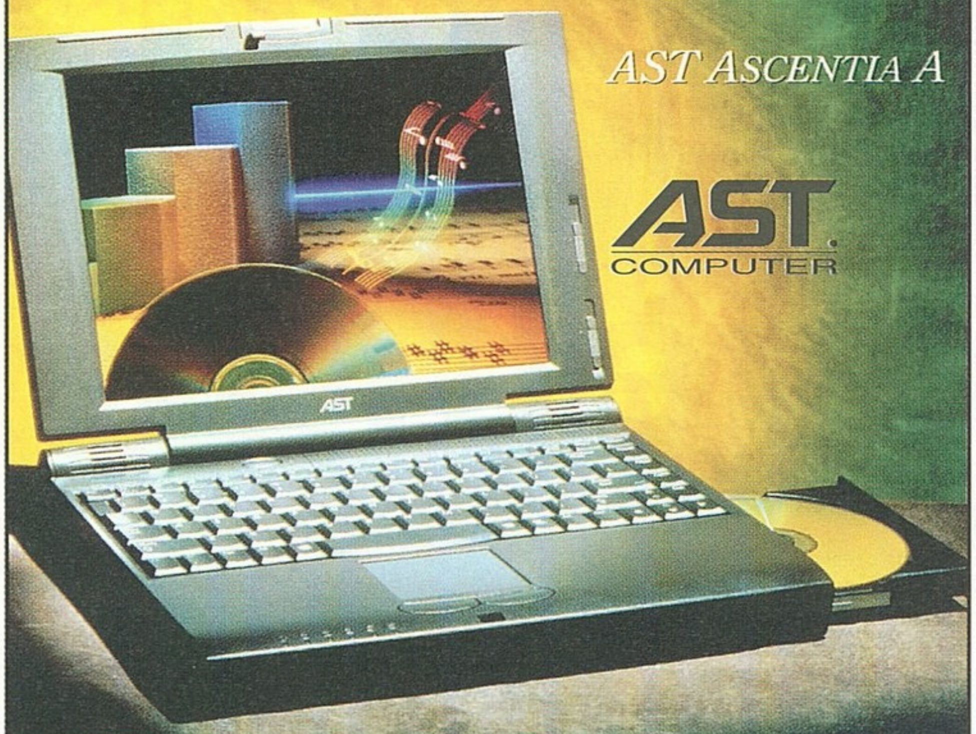
A Borland C++ Builder méltó párja lehet a pascalos világban gyorsan népszerűvé vált Delphinek, és ahhoz hasonlóan különböző „kiszerelésekben" kerül forgalomba is (Standard; Professional; Client/Server), továbbá több közös eszközt tartalmaz a Borland C++ legújabb verziójával.

Simay Endre István



Kapcsolat
a jövővel..
3Com
Látogasson meg partnereinknél
az IFABO-n
A pav. 209
ALBACOMP
A pav. 202/a
BCN
A pav. 306/a
COMPUTER 2000
33. pavilon
MŰSZERTECHNIKA
Computer Rt.
A pav. 306/e
NETWORX
Számítógéphálózatok Kft.
3Com Hungary, H-1036 Budapest, Lajos u. 48-66.
Tel.: 36-1-250-83-42, Fax: 36-1-250-83-47
http://www.3com.com
INFORMÁCIÓKÉRÉS: 050

A Legtöbb Érték És Minőség 450.000,- Ft Ért



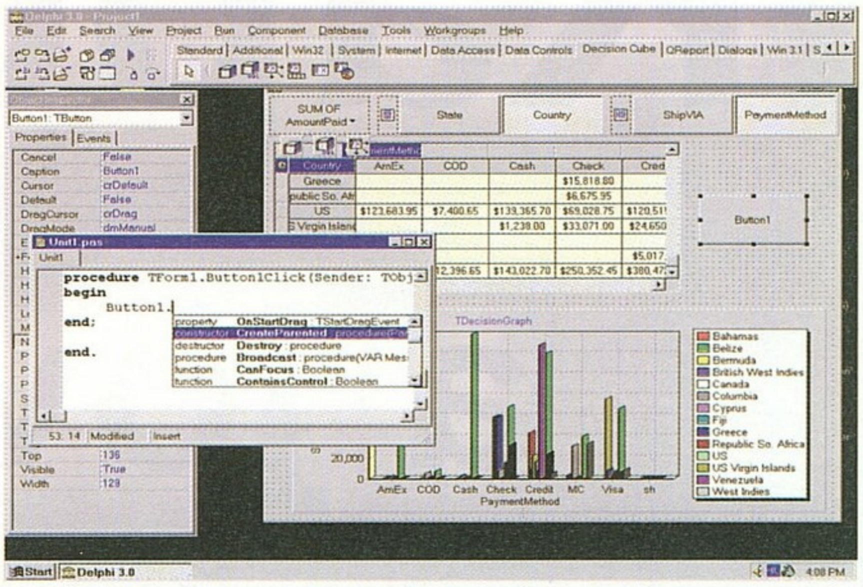
G70 Kft., az AST magyarországi disztribútora
1112 Budapest, Péterhegyi út 98. tel.: 228-4838

Szerződött partnereink:

AT-MDS Kft. Pomáz t: 06-26 325-054
B.I.L.L.-C Kft. Budapest t: 135-0550
Castan Bt. Orosháza t: 06-68 311-952
Compatibil Kft. Zalaegerszeg t: 06-92 311-100/14
Consultronics Mo.-i Képv. Budapest t: 275-1859
Delphi-Soft Kft. Budapest t: 265-4593
DNN Magyarország Kft. Budapest t: 135-4748
Duna Computer Százhalombatta t: 06-23 358-785
Ega-Trade Kft. Szeged t: 06-62 494-152
Enno-Sys Kft. Budapest t: 326-8621
FairSoft Kft. Miskolc t: 06-46 412-155
FEFO Kft. Budapest t: 267-8980
Flag Kft. Szombathely t: 06-94 322-134
Futurecom Kft. Budapest t: 212-0987
HAVE Kft. Debrecen t: 06-52 412-857
Ideal 2000 Kft. Budaörs t: 06-23 440-158
Intercas Kft. Budapest t: 155-2180
J.S.F. Kft. Szeged t: 06-20 460-742
Kürt Kft. Budapest t: 228-5410
Logosz Bt. Szeged t: 06-62 310-671
Prajzsnár Kft. Salgótarján t: 06-32 417-244
Provicon Kft. Budapest t: 129-6998
Quasar 2000 Kft. Budapest t: 111-4309
Start Up Kft. Zalaegerszeg t: 06-92 315-618
Summacomp Kft. Szeged t: 06-62 477-582
Szolex Kft. Nyíregyháza t: 06-42 451-605
Ten Technik Kft. Budapest t: 06-20 452-180
Traco-D Kft. Debrecen t: 06-52 431-297

INFORMÁCIÓKÉRÉS: 0510 ▲

DELPHI 3.0



* Package compiler technológia, az EXE-k mérete akár 20 kB-ra csökkenhet
* Decision Cube, többdimenziós adatelemzés grafikonok támogatásával
* Többrétegű alkalmazások fejlesztése a "Multi-tier Broker Architecture" segítségével
* A gépelés mennyiségének csökkentése a Code Insight-tal

Bemutató:
1997. május 21.
Liget Kongresszusi Központ
Budapest, Dózsa György út 84/a
Kezdés: 10 órakor
A részvétel mindenki számára ingyenes.

Jelentkezés: telefonon, faxon, vagy E-Mailen: delphi@delphi.hu
További információk web oldalunkon: www.delphi.hu

SilVert

Borland Delphi-Szoft **Delphi-Szoft**
1143 Budapest, Hungária krt. 79-81.
Telefon: 252-8145, fax: 252-8773
Internet: http://www.delphi.hu

INFORMÁCIÓKÉRÉS: 0512 ▲

SHARTECH COMPUTER

1087 Budapest Luther u. 1/c.
Tel.:114-0590, Fax:173-1809

TEAC 4x CD-író

4x írás / 4x olvasás
Hozzáférés: 220ms
Puffer: 1MB

16x!

TEAC CD-olvasó

16x olvasási seb.: 2.4MB/s
Hozzáférés: 105ms
Puffer: 128KB

160W hangfal

TEAC Power Max

2-utas sztereó hangfal
Beépített erősítő + táp 220V
50-20 000 Hz

ADATVISSZAÁLLÍTÁS

Vírusok, trójai programok, áramkimaradás vagy egyéb ok miatt elveszett adatok helyreállítása.

INFORMÁCIÓKÉRÉS: 0553 ▲

Nyelvoktató és oktató CD-k

A PICDIC sorozat, az Angol Kiejtésiskola, a ClipDIC English Part 1-2, a ClipDIC Deutsch 1, az Üzleti Angol 1, a ManóKaland és a ManóAngol után

MEGJELENT:

ManóNémet
5400 Ft+ÁFA

ClipDIC Deutsch 2.
6400 Ft+ÁFA

Üzleti Angol 2.
6400 Ft+ÁFA

Május 1-31 között minden kiadványunk 10 % kedvezménnyel vásárolható meg.

IFABO BUDAPEST 1997 *Szeretettel várjuk standunkon!* B PAVILON 11/H

Profi-Média Kft.

Profi-Média Kft.
6500 Baja,
Déri Frigyes sétány 4.
Tel./fax: 79/325-467
pmed@mail.matav.hu

INFORMÁCIÓKÉRÉS: 0545 ▲

Rövid hírek az OS/2 világából

Norloff listája

Az OS/2-es BBS-éről (http://www .os2bbs.com) világszerte ismert Pete Norloff (pnorloff@os2bbs.com) most új fába vágta a fejszéjét. Beindított ugyanis egy levelezési listát, amely az arra feliratkozóknak minden nap elküldi a BBS-ére bekerülő új fájlok részletes listáját. A feliratkozást a lyris@bbs .os2bbs.com címre küldött subscribe <listanév> tartalmú levéllel lehet megtenni, ahol a <listanév> helyébe az os2new-1day-html, os2new-1day-text, os2new-1week-html, os2new-1week-text kulcsszavak valamelyikét kell behelyettesíteni. Attól függően, hogy naponta vagy hetente, illetve hogy milyen formátumban kívánjuk a fájllistát megkapni. Szintén a felhasználók dolgát igyekeznek megkönnyíteni a nagy OS/2-es ftp site-ok üzemeltetői. Az ftp.leo.org-ra bekerülő új fájlok listáját is rendszeresen postázzák, azonban nem közvetlenül a felhasználókhoz, hanem a comp.os.os2.announce hírcsoportba. A hobbes.nmsu.edu pedig keresőrendszerrel gazdagodott: http://hobbes.nmsu.edu/cgi-bin/search. (Magyar OS/2-es keresőrendszer: http://xenia .sote.hu/cgi-bin/search-os2-archives)

Templeton 1.900

Az Interneten online módban eltöltött időt (és egyúttal telefonszámlánkat) csökkenthetjük a Neal Krawetz-féle (dr.neal@worldnet.att.net) Templeton programmal, amely komplett WWW site-ok letöltésére alkalmas. A program sokoldalúan konfigurálható, és automatizált letöltéseknél is jól használható. Egyetlen hátránya, hogy shareware, így regisztrálni kell a próbaidő letelte után. A beszerzés és regisztrálás a BMT Micro WWW oldalain keresztül (http://www.bmtmicro.com/catalog/tton) egyidejűleg lehetséges.

Az IBM SearchManager/2 bővítése

Ha nem találunk egy fájlt a merevlemezen, ráadásul még a nevére sem emlékezünk, akkor nagyon jól jön az IBM által kifejlesztett SearchManager/2. A program segítségével gyorsan behatárolhatjuk, hogy hol rejtőzik a keresett állomány. Ehhez az alkalmazáshoz készített egy kiegészítő DLL-t Steffen Siebert (http://www.cs.tu-berlin.de/~siebert), amellyel többek között a HTML vagy Gzippel tömörített fájlokra is kiterjeszthető az SM/2 hatóköre. A DLL az ftp://isis.sote.hu/pub/hobbes/diskutil/sm2addon.zip fájlban található. A SearchManager/2-ről a http://www.software.ibm.com/data/search/eyxn0mst.html oldalon olvashatunk bővebben.

DOS-játékok beállításai

A legtöbb DOS-játék futtatható OS/2 alatt is, ha ismerjük a játék számára optimális beállításait a DOS-szekcióknak. Nagyon sok beállítást találhatunk meg egybegyűjtve az OS/2 DOS games list oldalon (http://www.ifx.net/~crcomer/os2games), amely nemrégiben új helyre költözött, és jelentősen meg is újult. A szerkesztést jegyző Chris Comer (crcomer@home.ifx.net) szívesen vesz bármilyen megjegyzést vagy kiegészítést.

Új programok

Az OS/2-es programfejlesztők továbbra is igen aktívak. A szinte már minden hónapban rendszeresen megjelenő új PMMail mellett a PMDiff 4.0 fájl-összehasonlító program béta-verzióját (http://www.arac.demon.co.uk/betas.htm), az FTP Browser (ftp://isis.sote.hu /pub/hobbes/internet) és FTP-It! (ftp://ftp.wilmington.net/bmtmicro) 1.6 illetve 2.3-as változatát vehetik birtokba a felhasználók. Egy új grafikai program (Embellish) is fejlesztés alatt áll, amelynek béta-változatát már ki is próbálhatjuk, ha letöltjük az ftp://ftp.dadaware.com/dadaware/emb_os2.zip fájlt. 2-8 éves gyerekek számára készült az AlphaPro nevű játékprogram (http://www.bmtmicro.com/catalog/alphapro), amely a betűkkel és számokkal való ismerkedést segíti elő. Sajnos a programnak egyelőre csak angol és spanyol nyelvű változata van.

Team OS/2 FAQ 2.41

A Christian Alice Scarborough (ccs95@aber.ac.uk) által szerkesztett Team OS/2 FAQ-nak megjelent az újabb kiadása. Chris elfoglaltságai miatt ez az utolsó verzió, amelyet még ő készített. Az eredeti, angol nyelvű változat a http://www.teamos2.org/faq, a kis késéssel készülő magyar nyelvű fordítás pedig a http://xenia.sote.hu/os2/teamos2/team_os2.html oldalon található.

Lyris levelezőlista

A Walter Shelby csoport piacra dobta OS/2-n is futó levelezőlista programját. A készítő cég információs anyaga külön kiemeli a beépített „e-mail-motor" nagy sebességét (több száz levél percenként) és a rendszer biztonságosságát. További információ olvasható a Lyris honlapon: http://www.lyris.com. (Nem lehet rossz program, ha a Norlofték is ezt használják! - A szerk.)

Az OS/2 jövője

A Hannoverben megrendezett CeBIT-en készített interjút a német TEAM OS/2 John Soyringgal, az IBM vezető beosztású alkalmazottjával. A beszélgetés során Soyring elmondta, hogy az IBM még legalább tíz évig, tehát minimum 2005-ig fogja fejleszteni és támogatni az OS/2-t. Ugyanakkor azt is elismerte, hogy az IBM sorozatosan hibákat követett el az OS/2 menedzselése során. Példaként említette a klienskernel fejlesztés leállításának bejelentését, amelyet sokan hibásan úgy értelmeztek, hogy az OS/2 fejlesztésének befejezését jelenti. Az ilyen esetek elkerülése érdekében az IBM a jövőben 90 naponként informálni fogja a sajtót az OS/2 fejlesztésének aktuális eredményeiről. Soyring elmondása szerint az IBM tavaly 150 millió dollárt költött az OS/2 fejlesztésére, és az elmúlt év első kilenc hónapjában több kódot írtak, mint az azt megelőző 10 évben!

A fejlesztésre fordított összeg a jövőben sem fog csökkenni, mivel az OS/2-t igen intenzíven használja a világ sok nagyvállalata. Páldaként említette a Hong Kong Chinese Bankot, amely a közelmúltban írt alá megállapodást az IBM-mel arról, hogy gépeiken az MS Windowst OS/2-re cserélik. A banknak kb. egymilló alkalmazottja van, és ebből lehet következtetni az általuk használt PC-k számára is.

Kádár Zsolt

A KOSSUTH KIADÓ MULTIMÉDIA-FEJLESZTŐ PÁLYÁZATA AMATŐRÖK RÉSZÉRE

A Kossuth Kiadó pályázatot hirdet 18 éven aluli és 18–26 éves amatőr multimédia-fejlesztők számára, ismeretterjesztő (oktató, felvilágosító, játékos készségfejlesztő, játék stb.) program elkészítésére.

A pályázaton egyénileg és csoportosan is lehet indulni. Kizárólag amatőr pályázók jelentkezését fogadjuk el, akiknek munkaköre (vállalkozói tevékenységi köre) nem multimédia-fejlesztő. Egy pályázó több pályázattal is nevezhet. A pályázat jeligés, minden pályázatot más-más jeligével kell ellátni.

A pályamunkával kapcsolatos feltételek:
✦ tartalmazza a multimédia szokásos összetevőit (grafikus felület, szöveg kereszthivatkozásokkal, kép, hang, esetleg mozgókép, animáció) ✦ terjedelme minimálisan 350 MB ✦ átlagos hardverigény (486-os PC, 8 MB RAM) ✦ beadása CD-ROM lemezen, vírusmentesen (a vírusfertőzött anyagokat elbírálás nélkül visszaküldjük).

A pályázatra benyújtott programokat változtatás nélkül és a kitöltött jeligés pályázati űrlap alapján értékeljük.

A pályázat elbírálásakor előnyt jelent, ha a program:
✦ többnyelvű ✦ több platformú (PC + Macintosh) ✦ a témát játékosan dolgozza fel ✦ kérdésekkel, feladatokkal próbára teszi a program használata során megszerezhető ismereteket.

A legjobbnak ítélt pályamunkát a szerzővel kötött megállapodás alapján a Kossuth Kiadó megjelenteti.

Az első három helyezett pályamunkát értékes nyereményekkel díjazzuk.

A pályázat beadási határideje: 1997. augusztus 31.

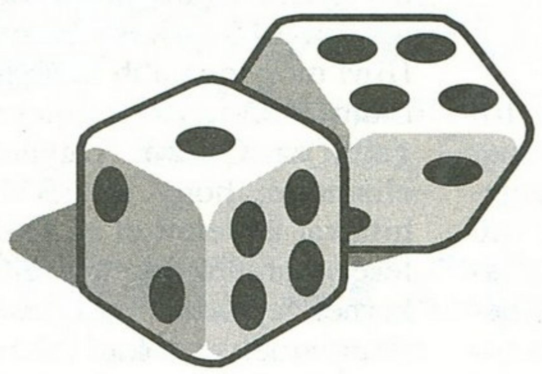
Eredményhirdetés: 1997. szeptember 30.

Pályázati űrlap és további tájékoztatás a következő címen kérhető:

Kossuth Kiadó Rt.
Multimédia Szerkesztőség
1325 Budapest, Pf. 244 • Telefon: 370-0608

INFORMÁCIÓKÉRÉS: 0525 ▲

Ne kockáztasson!



VirWare
vírusvédelem

...és a biztonság visszatér
forródrót: (20) 421-174
e-mail: wirware@mail.datanet.hu

megjelent a 7.0 verzió
DOS, Windows 3.x, Win95, Windows NT

array

Az ASK megmutatja az IMPRESSION 8300-as projektorát

- Felbontás valós 800 x 600 (komprimált 1152 x 870)
- 16,7 millió szín
- 450 ANSI lumen
- Motoros ZOOM
- S-Video, NTSC, PAL
- 2 x 5 Watt sztereó hang
- 7,7 kg
- Szuper csendes ventillátor

Az NSM új Jukebox konfigurációja

- Moduláris felépítés
- 60–135 CD-kapacitás
- 40–90 GB adatmennyiség
- Max. 5 drive + a mailslot
- Plextor 12x CD-ROM
- Yamaha 480 4x/6x CD-író
- SCSI-2 adatinterfész
- RS 232 robotinterfész
- iXOS manager software

Találkozzunk az IFABO-n!
A 309/B KFKI Direkt stand

Viszonteladókat keresünk!

array Data Hungária Kft.
1094 Bp. Balázs Béla u. 35.
Tel./Fax: 455-6892, 93, 94
E-mail: array_bp@mail.elender.hu

INFORMÁCIÓKÉRÉS: 0505 ▲

Az együttélés bonyadalmai

Öt operációs rendszer egy gépen

Többféle operációs rendszert installálni egyetlen harddiszkre nem mindig egyszerű. Cikkünk ebben szeretne segíteni, elkerülendő a buktatókat.

A „normális" felhasználó hosszú ideig csak egyféle (DOS) operációs rendszer alatt dolgozott. Legfeljebb a hozzám hasonló oktató-újságíró-hobbizó találta ki magának, hogy egyetlen gépére többféle operációs rendszert telepít és használ. Azután a rendszerválaszték növekedésével egyre többen kerültek hasonló helyzetbe. Például ki szerették volna próbálni az OS/2-t vagy a Win95-öt, de minden eshetőségre készen az előző rendszer megtartása mellett.

Nézzünk meg most egy konkrét lehetőséget. A következő operációs rendszereket akarjuk futtani az egyetlen harddiszkkel ellátott gépen: DOS, Windows 3.1, Windows for Workgroups 3.11, Windows 95, Windows NT Workstation 4.0 és OS/2 Warp.

Jól oszd fel a winchestert!

A winchester egymástól független, külön meghajtóként kezelhető területekre, ún. partíciókra osztható. Maximálisan négyre, mert ennyi bejegyzéshez van hely a partíciós táblában. Ez a tábla a harddiszk legelső szektorában található, az ún. masterboot szektorban az operációs rendszert betöltő rutinnal együtt.

A FAT fájlrendszertől induló operációs rendszerek (DOS, Windows, OS/2) kétféle partíciót ismernek, az eredetileg a rendszer indítására szolgáló elsődleges vagy primary partíciót, és a kiterjesztett vagy extended partíciót. Egy winchesteren alapértelmezés szerint csak egy-egy lehet mindkettőből. Az extended partíció további részekre, logikai meghajtókra osztható, amelyek szintén független meghajtóként viselkednek. A nem FAT gyökerű Unix rendszereknél nincs ilyen megkülönböztetés. Nézzük végig az ábrán látható konfiguráció kialakítását!

Az FDISK-kel létrehozott elsődleges DOS-partíció mérete csak 250 MB, mert 262 MB felett a harddiszk helyfoglalási egysége (a cluster) már 8 KB, vagyis a legkisebb fájl is minimum ennyi helyet foglal el, a mi esetünkben ez most 4 KB. A Windows 3.1 és a Windows for Workgroups (WFW) 3.11 különböző könyvtárakba telepítve használható, de a WIN parancsot csak az indítani kívánt változat könyvtárába való belépés után kell kiadni. A kényelmesebbek készíthetnek erre egy-egy batch fájlt is. Egyedül az állandó „kihelyezett rakterület" (a swapfájl vagy lapozófájl) okozhat bonyodalmat, mert ezt azonos néven mindkét rendszer a főkönyvtárban tartja, és a másik verzió által létrehozottat nem fogadja el. A megoldás többféle lehet, például ha van kiterjesztett DOS partíció a harddiszken, akkor az egyik kerüljön oda.

A mi esetünkben az egyiknél vagy mindkettőnél beállíthatunk ideiglenes lapozófájlt. Ha van elegendő RAM (legalább 16 MB), és nem túl nagyok az alkalmazások, akár nélkülözni is tudjuk a swapfájlt. Mindez a Vezérlőpult/386-os mód/Virtuális memória ablakban tehető meg.

A Windows 95 indulása...

A Win95 telepítésekor új könyvtárat megadva megtarthatjuk a 3-as verziókat. Ekkor az előző DOS-változatot is megőrzi (ki tudja miért, a Novell DOS 7-et nem), a rendszerfájlokat — IO.SYS, MSDOS.SYS, COMMAND.COM, CONFIG.SYS és az AUTOEXEC.BAT — DOS kiterjesztésre nevezi át. Az előző DOS-ra ritkán van szükség, mert a Windows 3.1 és a WFW 3.11 fut Win95 DOS alatt is. Fordítva nem igaz a dolog, a Win95 nem indítható el az MS-DOS 5.0 vagy 6.x alól.

A 3-as Windowsok természetesen nem indíthatók el sem a Win95 DOS-ablakában, sem pedig a Win95-ből való kilépéskor felkínált MS-DOS módban. Ehhez az is szükséges, hogy induláskor csak a valós módú rendszerfájlt, az IO.SYS-t engedjük betölteni, a védett módba átkapcsoló grafikus felületet nem. Ezután a szokásos DOS parancssorból pedig elindíthatjuk a 3-as vagy a 95-ös Windowst.

Erre az első lehetőség, hogy a „Windows 95 indulása ..." felirat megjelenésekor az F8 lenyomásával kérjük az indítómenüt, és itt csak a parancssor betöltését választjuk. Kényelmesebb módszer, ha a főkönyvtárban lévő, rejtett MSDOS.SYS szövegfájlba felvesszük a BootGUI=0 bejegyzést, mert ekkor nem indítja el a grafikus felületet, nem kell az F8-cal reflexgyakorlatot végeznünk.

A korábbi DOS-változat betöltésére csak akkor lesz szükség, ha meg akarjuk változtatni a Windows 3.1/WFW 3.11 lapozófájl beállítását, mert a Win95 a hosszú fájlnevek védelmében letiltja a közvetlen lemezhozzáférést. Az előző DOS betöltését szintén az indítómenüből választhatjuk, vagy pedig, ha a már említett MSDOS.SYS-ben szerepel a MultiBoot=1 bejegyzés, akkor a „Windows 95 indulása ..." feltűnésekor elég az F4-et lenyomni. Az előző DOS indításakor a Win95 rendszerfájlokat .W40 kiterjesztésűre nevezi át.

Új lakó

Az OS/2-t telepíthetjük a DOS partícióba, vagy létrehozhatunk számára saját külön partíciót. A DOS partícióba való telepítés esetén az OS/2 BOOT parancsával választhatjuk ki, hogy a következő rendszerindításnál a DOS-t vagy az OS/2-t töltse be. A parancs egyszerűen a főkönyvtárban lévő rendszerfájlokat másolgatja oda-vissza, a kért rendszernek megfelelően. Maga az OS/2 kézikönyv is azt ajánlja, hogy készítsünk inkább saját partíciót a rendszer számára, annál is inkább, mert azon használható a HPFS (High Performance File System) fájlrendszer. A HPFS előnye a FAT-tal szemben a gyors elérés, és hogy a fájlok sokkal kevésbé hajlamosak a széttöredezésre. Igaz, hogy a HPFS több memóriát foglal, és a lemezen nagyobb az adminisztrációs terület aránya, ezért 60 MB-nál kisebb partícióra nem ajánlott.

Amennyiben külön partícióra teszszük az OS/2-t, minden gépbekapcsolás után el kell dönteni, hogy melyik partícióról akarunk operációs rendszert indítani. Ezért van szükség az OS/2 betöltő programjára, a Boot Managerre. Ez egy 1 MB-os önálló partícióba kerül, és ezt a partíciót kell aktívnak, azaz rendszerindításra alkalmasnak megjelölni. Bekapcsolás után a Boot Manager jelentkezik be, és menüjéből kiválasztható az indítandó partíció.

Pusztán a partíció kiválasztásához nem lenne szükség a Boot Managerre. Amíg csak kétféle Unixot (Coherentet és Linuxot) használtam a DOS/Windows mellett, erre a célra tökéletesen megfelelt egy kis rutin, ami belefért a masterboot szektorba. Induláskor vagy az aktívnak megjelölt partíciót töltötte be, vagy ha ilyen nem volt, akkor a partíció számát kérve indította el a kívánt rendszert. A Boot Managerre a már említett elsődleges/kiterjesztett partíció megkülönböztetése miatt van szükség.

Az OS/2-t telepíthetjük elsődleges partícióra, vagy a kiterjesztett partíció egyik logikai meghajtójára. Az alapelv szerint a rendszernek csak egyetlen elsődleges partíciója lehet, ha az OS/2 telepítőjével létrehozunk még egyet, akkor a betöltéskor a másikat rejtetté kell tenni, vagyis egy olyan azonosítóval ellátni, amit a rendszer nem ismer.

A Boot Manager tehát egyrészt kiválasztja az indítandó elsődleges partíciót, és mindig ráteszi a rejtett jelzést a másikra. Ez azt jelenti, hogy ha betöltjük az OS/2-t, akkor az a DOS partíciót nem fogja látni, az ott lévő adatokhoz és programokhoz nem lehet hozzáférni. A DOS betöltésekor ez nem olyan nagy baj, a DOS ugyanis a HPFS fájlrendszer miatt amúgy sem látná az OS/2 partícióját.

Célszerűbb tehát az OS/2-t nem elsődleges partícióra, hanem logikai meghajtóra telepíteni. Kiterjesztett partícióból az alapértelmezés szerint nem lehet rendszert indítani, a Boot Manager ezért most sem hagyható el. A telepítés három fő lépésből áll, és általában nem okoz különösebb gondot:

1. Telepítsük a Boot Managert.
2. Hozzunk létre egy logikai meghajtót az OS/2-nek.
3. Az OS/2 és a DOS/Windows partíciót vegyük fel a Boot Manager indító menüjébe.

A Boot Manager paraméterei később az OS/2 FDISK parancsával bármikor megváltoztathatók. Van olyan shareware DOS-segédprogram, amellyel olvasható a HPFS partíció.

Na még egy rendszert!

A januárban megjelent magyar Windows NT Workstation 4.0 erőforrás-igényes, de jó rendszer, a kipróbálási lehetőséget a Microsoft Magyarországnak köszönhetjük. Az NT-t telepíthetjük az elsődleges partícióba, de a korábbi Windowsok futtatásáról le kell mondanunk, vagy pedig a kiterjesztett partíció egyik logikai meghajtójára. Az NT a FAT rendszeren kívül saját NTFS (NT File System) fájlrendszert is használhat. A már a HPFS-nél említett előnyökhöz járul a fájlszintű tömörítés, amelyet az NT az NTFS rendszeren automatikusan kezel. Külön logikai meghajtóra telepítésnél mindenképpen érdemes az NTFS-t választani, egyetlen hátránya, hogy ezt a meghajtót a DOS/Windowsból nem fogjuk látni. Némileg ezen is lehet segíteni, van olyan shareware segédprogram, amellyel DOS alól olvasható az NTFS fájlrendszer.

Az NT telepítéséhez először létre kell hozni számára egy logikai meghajtót a DOS alól. Ezt a DOS csak FAT-ra tudja formázni, majd az NT konvertálja NTFS-re. A logikai meghajtóról nem tudnánk elindítani az NT-t, ezért az elsődleges DOS partícióban lecseréli a bootszektort, és a főkönyvtárban elhelyezi a saját betöltőjét, az NTLDR-t. Ez indul majd el, és megkérdezi, hogy az MS-DOS-t vagy az NT-t kívánjuk-e elindítani.

Mire kell ügyelni?

Fontos a sorrend, vagyis először az OS/2, és csak azután az NT. Ellenkező esetben ugyanis az OS/2 valamilyen okból saját HPFS formátumának ismeri fel az NTFS-t, és abba lelkiismeretfurdalás nélkül beleír, az NT-t ezek után már nem tudjuk elindítani. Az OS/2 telepítése előtt kell létrehoznunk a DOS-ban két logikai meghajtót, egyet az OS/2, egyet pedig az NTFS számára. Ha ezt nem tesszük meg, az OS/2 telepítője csinál magának egyet, de az NT számára később már nem tudunk még egyet létrehozni: a DOS FDISK-je azt üzeni, hogy már van egy kiterjesztett partíció, még egyet nem csinál.

Az OS/2 rendszer partíciója nem nyúlhat át az 1024 cilinderes határon a harddiszken. Lehet ezen túl is partíciót használni, a korlát csak a rendszerfájlokat tartalmazó partícióra érvényes.

Az NT a telepítés során újraindítja a rendszert. Az elsődleges partíciók között — ha az OS/2-t mégis erre és nem a logikai meghajtóra tettük — a Boot Manager rakosgatja a rejtett jelzést. Az NT telepítője felismeri a Boot Managert, és a telepítés során letiltja. Később persze újra lehet engedélyezni. Ha azonban az NT telepítése előtt az OS/2-t használtuk utoljára, akkor a rejtett jelzés a DOS partíción maradt, és az NT az OS/2-t indítja újra a telepítés alatt. Az eredmény teljes káosz.

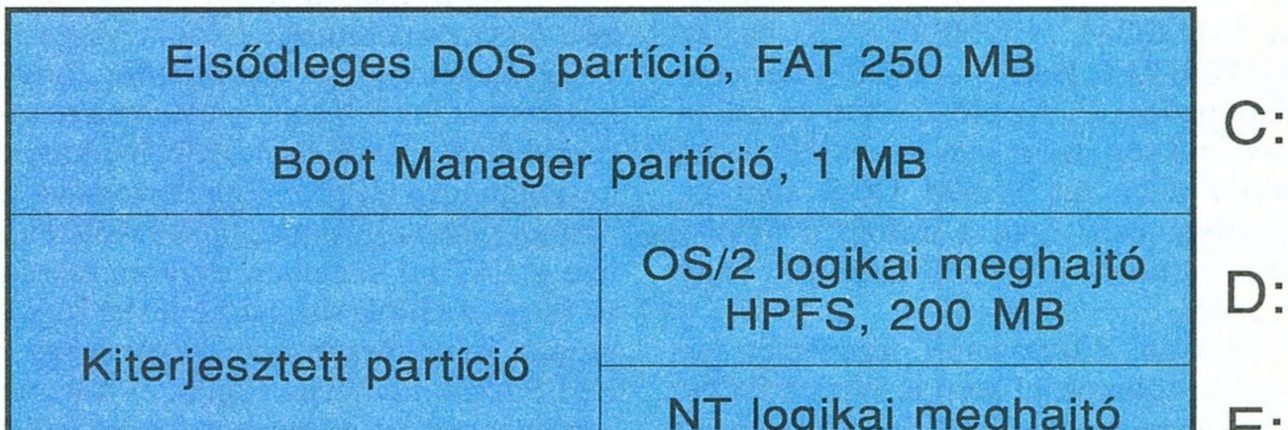
Végül tekintsük át, hogyan működik a kész konfiguráció.

1. A Boot Manager partíciója van aktívnak kijelölve, a masterboot tehát ezt indítja el. A Boot Manager menüben két bejegyzés szerepel, a DOS/Windows elsődleges partíció, és az OS/2 logikai meghajtója. Az OS/2-t választva indul a Warp, és elérheti a DOS/Windows partíciót is, az NTFS-t természetesen nem (pontosabban a meghajtót üresnek látja).

2. A DOS/Windows partíciót indítva először az NT betöltője jelentkezik be, és megkérdezi, hogy az NT-t vagy pedig az MS-DOS-t akarjuk-e.

3. Az MS-DOS-t választva jelenik meg a „Windows 95 indítása ...” felirat, és a CONFIG.SYS menüjében választhatjuk ki a Win95-nek, a Windows 3.1-nek vagy a WFW 3.11-nek megfelelő konfigurációt. Mivel a BootGUI=0 beállítással letiltottuk a Win95 grafikus felületének automatikus indítását, a szokásos DOS-parancssort kapjuk, ahonnan egy-egy batchfájllal indítható a három közül bármelyik Windows vagy valamilyen DOS program.

Csórián Sándor



DOS doktor II.

A nélkülözhetetlen fájlkezelők

Folytatjuk a múlt hónapban megkezdett sorozatot. A most következő rész a talán leginkább elterjedt két fájlkezelővel foglalkozik: a Norton Commanderrel és a DOS Navigatorral. Nem annyira bemutatásról van szó, hiszen mindkettő régóta ismeretes és széles körben elterjedt a számítógépet használók körében, hanem a kettő összehasonlításáról, erényeikről és hibáikról.

DOS alatti munkánkat jelentősen megkönnyítik — gyorsabbá, kényelmesebbé teszik — a különféle fájlkezelők: Norton Commander, DOS Navigator, Volkov Commander, DOS Shell és társaik. Mint más programok esetében, ezeknél is több verzió használatos, és az újabbak tudásszintje, szolgáltatásaik minősége egyre jobb, mert folyamatos fejlesztés alatt állnak.

Ez a mostani írás leginkább azokhoz szól, akik használják a két legelterjedtebb közül az egyiket vagy a másikat, esetleg mindkettőt. Természetesen aki még csak most ismerkedik a számítógép használatával, itt az is hasznos tanácsokat kaphat.

Ne becsüljük le a kezdőket, egykor valamennyien azok voltunk, s a kezdők közül nőnek ki a profik. Honnan, ha nem közülük? Másrészt általános tapasztalat, hogy hiába használ valaki régóta valamilyen programot, az még nem jelenti a program kielégítő ismeretét is. Ráadásul sokan csak shareware verziókkal dolgoznak, és ezekről általában nincs is részletes leírás, vagy ami van, az is csak angolul. Nézzük hát először áttekintésként azt, ami ezekben a programokban közös.

Ami közös bennük

— *Zseniálisan megszerkesztett* programok: áttekinthetőség, precizitás, könnyű kezelhetőség, könnyű tanulhatóság, sokrétűség, gyorsaság tekintetében szinte felülmúlhatatlanok, egyedülállóak. A Windows alá írt utánzataikkal összevetve pedig összehasonlíthatatlanul jobbak!

— *Karakteres felhasználói felület* a jellemzőjük, némi grafikus elemmel vegyítve, amilyenek például a funkciógombok. Mindkettő lehetővé teszi, hogy az ablakok mögé nézve a DOS üzeneteit elolvashassuk.

— *Teljes képernyős, interaktív* felhasználói felülettel rendelkeznek.

— *Két különböző meghajtó* adatait tudják megjeleníteni egyidejűleg, külön ablakokban. Alkalmasak továbbá a két különböző meghajtó teljes fájladatait egyszerre láttatni, az eredeti karakterméretben, tehát az olvashatóságot nem rontva.

— Képesek *az összes fájlművelet végrehajtására*. Törlés, másolás, áthelyezés, átnevezés, létrehozás, szerkesztés, összehasonlítás, összefésülés, betekintés stb.

— Alkalmasak *a fájladatok osztályozására* különböző szempontok szerint: név szerint ábécésorrendben vagy fordítva; kiterjesztés szerint; méret szerint; a létrehozás, illetve a módosítás időpontja szerinti rendezésre le- és felléptető sorrendben.

— *Lehetővé teszik az egérhasználatot*. Dupla kattintás, az újabb verziókban „fogd és vidd" funkció, kurzorpozicionálás, gombok használata, kijelölés, menük aktivizálása stb.

Pro és kontra

Nézzük meg azt is, mi az, amiben különböznek! (A tömörség kedvéért a továbbiakban a DOS Navigator helyett a DN, a Norton Commander helyett az NC rövidítést fogom használni.)

Tökéletes fájlkezelő program sajnos nincsen. Addig, amíg a fejlesztők nem látják szükségesnek a legjobb programok legjobb tulajdonságainak összehozását egyetlen programon belül — és ez sajnos még jó ideig nem várható, pedig sok cég tett már ilyen ígéretet, de valamit mindig „kifelejtenek" —, addig kénytelenek leszünk hasonló funkciókat ellátó, mégis különböző programokat egymás mellett megtartani. A mindinkább egyeduralkodóvá váló Windows-verziókon belül sem bukkantak még fel olyan fájlkezelő programok, amelyek teljes egészében nélkülözhetővé tennék ezt a két fájlmenedzsert...

Nos, az első és legfontosabb különbség a memóriahasználatban van. A DN mint memóriarezidens program 1 Kbájt memóriát foglal le. Ezzel szemben az NC hatszor(!) ennyit, azaz 6 Kbájtot. Mivel a két program közel azonos tudású, a DN programozói sokkal szigorúbbak voltak önmagukhoz, és jobban ügyeltek az erőforrások gazdaságos felhasználására, mint a NC programozói.

Egy másik szempont az NC javára szól: a panelbeállítások könnyebben elérhetők és megváltoztathatók, mint a DN-ben, egyszerűbben meghatározhatjuk, hogy milyen rendszerezésben kérjük a fájlinformációkat (név, kiterjesztés, méret szerint).

Amiben megint a DN jobb: az ablakok alatt megjeleníthetjük a meghajtókat, és ezek közül egérrel is választhatunk, a kijelölt meghajtó állományait pedig láthatjuk az illető ablakban.

A következő különbség szintén a DN felé billenti a mérleg nyelvét: a választott meghajtó alatt folyamatosan megjeleníthetjük a meghajtón rendelkezésre álló szabad hely méretét.

A „mindenható" F3 viszont nagyon is az NC mellett szól. Az NC „nézőkéje" sokkal rugalmasabb a DN „nézőkéjénél", mert mindenfajta fájlba betekinthetünk vele, és közel húszféle szövegszerkesztővel előállított állományt konvertálhatunk olvashatóvá. Az F3 funkció képes a JPG, BMP, TIF, GIF formátumú képek megjelenítésére is (az új változatban a GIF sajnos már nincs benne), és ezen belül a nagyítás, kicsinyítés, forgatás, színkonvertálás lehetőségét is biztosítja.

Egy másik érv a DN mellett: mód van az ASCII-táblázat lehívására, így speciális karakterek megadására is lehetőségünk nyílik, amilyenek például a táblaelemek.

Ugyancsak a DN pozitívuma: az Alt+G lenyomásával a kurzorvonalon

lévő könyvtár terjedelmét megnézhetjük, de akár több kijelölt könyvtárét is, sőt lehetőségünk van a könyvtárak folyamatos megmérésének beállítására. Ez utóbbit viszont nem ajánlom, mert jelentősen lelassítja a fájlkezelő működését, még egy Pentium 133-as, 24 Mbájt RAM-mal futó gép esetében is.

A keresési opció összehasonlítása az NC mellett szól: keresője sokkal összetettebb, jobban használható, mindenféle opciót beállíthatunk rajta, megadott dátum előtti és utáni, megadott méret alatti és feletti stb. fájlokat kereshetünk. A DN keresője gyorsabb ugyan, de nem ilyen sokoldalú.

A DN archívkezelője viszont kifejezetten intelligens, így például a Shift+F2 lenyomásával azonnal ki tudunk csomagolni egy tömörített fájlt, és a DN automatikusan tudja, hogy milyen eljárással kell kicsomagolni — persze, ha gépünkön ott vannak a megfelelő tömörítö-kibontó programok. Ugyanilyen könnyű a becsomagolás is, a Shift+F1 lenyomásával menüből választhatunk a rendelkezésünkre álló tömörítőprogramok közül.

Egy kényelmi funkció: igaz, hogy a panelbeállítások könnyebben elérhetők az NC-ben, a DN-ben viszont az Alt+Del megmutatja, milyen kiterjesztésű fájlok találhatók az aktuális alkönyvtárban, és ezek közül kiválaszthatjuk azokat, amelyeket látni szeretnénk.

A DN legújabb verziójában CD-lejátszót is találunk, melynek kezelése igen egyszerű, s lehetőségünk van audio-CD hallgatására munka közben is.

A DN-ben találhatunk többek között számológépet, játékot, sőt telefonkönyvet, amelyből tárcsázni lehet a kiválasztott számot, ha van modemünk.

A DN-ben egyszerre kettőnél is több ablakot nyithatunk, ami akkor előnyös, ha sok meghajtóval rendelkezünk.

A legújabb DN-verzió szövegszerkesztőjében, ha például az autoexec szerkesztésénél egy sort kiiktatunk a REM paranccsal, akkor ez a sor halványabban fog megjelenni, mint a többi, továbbá beállíthatjuk a zárójelek páros kiírását, így ha zárójelet kezdünk, nem kell ügyelnünk arra, hogy bezárjuk. (Természetesen a zárójelek bármelyike külön-külön törölhető hagyományos módon.)

Akkor hát melyiket ajánlom?

Mindkettőt szeretni lehet — egyszerre is. A két program együttesen alig több, mint 4 Mbájt helyet foglal el a winchesteren, és ennyi helyet még egy igen kicsi, 200 megás lemezen is érdemes áldozni rájuk.

Nekem két gépem van: egy 386 SX 33-as, 4 MB RAM-mal és 400 MB-ra feltuningolt 200 MB-os winchesterrel, valamint egy igazi „erőgép", Pentium 133-as, 24 MB RAM-mal, 1,3 GB-os winchesterrel. Mindkét gépen fel van téve a PATH-ra ez a két program. Elsődlegesen a DN-t használom, de gyakran behívom az NC-t is, amikor az jobban megfelel céljaimnak. A DN-t egyébként úgy állítottam be, hogy megjelenésében teljesen olyan legyen, mint a Norton Commander — igen, a sokféle egyéb beállítás mellett ez is lehetséges!

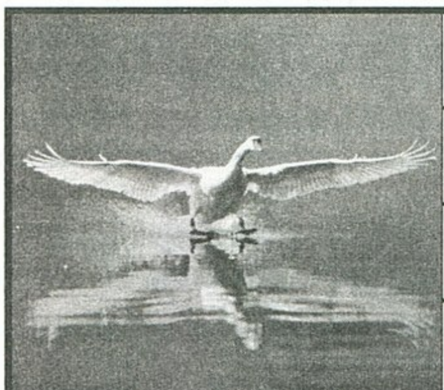
Végezetül hadd hívjam fel a figyelmet két súlyos fogyatékosságra:

1. Bármilyen praktikus volna, és a józan ész is azt diktálná, sem a DN, sem az NC nem teszi lehetővé, hogy amennyiben 1,44 MB-nál nagyobb terjedelemben kívánunk másolni egy floppyra, akkor a program kérje a következő lemezt, pedig ezt programozástechnikailag könnyű megoldani. A Windows kezeli ugyan ezt a problémát, de csak az egyik irányban: kéri a következő lemezt, de lemezről történő másoláskor ugyanezt már nem teszi meg...

2. Ha floppylemezre másolunk, és azon már van valamennyi adat, akkor a másolás megszakad a „Disk full" üzenettel. És ez nemcsak DOS alatt következik be, hanem a „fejlettebb" Windows alatt is. Száz bájt is elegendő volna a programban arra, hogy ilyenkor kiírja „Ennyi meg ennyi bájt van a lemezen. Felülírjam? I/N". Érdekes módon más programok megteszik ezt, még bizonyos Microsoft-termékek is...

Lehet, hogy igazságtalan vagyok, de csak azzal tudom magyarázni a fenti gyengeségeket és fogyatékosságokat, hogy a szoftverfejlesztő cégeknek nem áll érdekükben egy programon belül mindent megoldani. Akkor ugyanis holnap mit adnának el nekünk?

Mózes István Miklós



Azonban
Hét naponta ismétlődő
Féltőn védelmez
Tengeri rabló
Oroszlán becézve
MEGFEJTÉS II.
Község Csongrád m.-ben
Új Alaplap
Üvegfényű ásvány
Lopva eltanul
Argentína röv.
Aromás ital
Misike
..... Béla (Az MTK játékosa)
Félezer!
Fül, angolul
MEGFEJTÉS I.
Női név
Fásított nyilvános park
A Dollart-öbölbe ömlő német folyó
Patás állat
Angol tagadás
.....-bong
Helytelenít
Klubigazolványa van
Írófolyadék
Gondolatokat előszóban közölt
Hódító .....
Nagy befogadóképességű
Tágas helyiség
Európium vegyjele
Rangjelző előtag
Irország, írül
Mód szélei!
Lásd ott, röv.
Német folyó
Szintén ne
Adnak neki
Roald ...... (norvég felfedező)
Bolt
Iratot továbbít
Egyik északi nyelv
T.A.A
Igazgató, tréfás szóval
Határozott névelő
Szántóeszköz
Palindrom férfinév
Kettőzve: kedvelt cukorkaféle!
Jorge ...... (brazil író)
Álcáz
Étkezési célokra is használják
Telesport, röv.
Nagy-Britanniában élő
Női név
.... Kilimandzsáró hava
Cégforma, röv.
Disznó kemény, merev szőre
Sándor becézve
Holland Királyi Légitársaság

Hány E betűt rejt a megoldás?

Hívja az
PHONE CONTACT
rejtvényvonalát a
(06-90) 358-632-es
telefonszámon!

E havi rejtvényünk száma: **11**

A helyes megfejtők között egy 2 személyre szóló repülőjegyet sorsolunk ki az Amszterdam-Budapest-Amszterdam útvonalra a KLM jóvoltából!

1. Fejtse meg a rejtvényt!
2. Keresse meg a választ a kérdésre!
3. Hívja a megadott telefonszámot!
4. A telefon nyomógombjai segítségével adja meg a kérdésben megjelölt betűk számát, majd hagyja meg nevét, címét, telefonszámát!

Phone Contact Kft. A hívás díja 88 Ft+áfa percenként

Gráfok és hálózatok kezelése számítógéppel II.

A grafikus megjelenítés

Az első részben bemutatott alapelvekhez, tárolási formákhoz már mellékeltünk egy leegyszerűsített megjelenítési eszköztárat is, bízva abban, hogy felkeltjük vele az érdeklődést. A most tárgyalandó grafikus megjelenítés „csak” annyival tud többet az első részhez mellékelt megjelenítéshez képest, hogy míg ott csak a képernyőre eső koordinátákkal dolgoztunk, addig itt „tetszőleges” koordinátákkal. Az idézőjel azonban jelzi, hogy mivel számítógépes megoldásról van szó, azért valamilyen határokat szabnunk kell majd a gráfpontok koordinátáira vonatkozóan. Példánkban mi 6 jegyű e gész számok használatát engedtük meg.

Aki figyelemmel kíséri majd az elkövetkezendő, talán kicsit nehezebb, de mindenképpen érdekesebb, s gyakorlati szempontból is jól használható fejezeteket, bizonyára nem bánja meg, hogy elkezdte az alapozást.

Körvonalazzuk a most ránk váró feladatokat. Nevezetesen:

Az eredeti/fizikai koordinátákat transzformálnunk kell a képernyőn elhelyezkedő, a rajzolásra fenntartott területre (rajzterület). Ez a transzformáció legyen aránytartó, azaz ha a pontokkal például Magyarország településeit, az élekkel pedig a közutait modellezzük, akkor a képernyőn egy méretarányos Magyarország jelenjen meg!

Nagy hálózatok/gráfok esetén, mivel sok a pont, és viszonylag kis helyen kell őket megjeleníteni, elkerülhetetlen a pontok „összezsúfolódása”. (Például az előbbi „térképen” Budapest és környéke, feltéve persze, hogy hálózatunk kellően részletes, és nem csupán egyetlen pont szimbolizálja a fővárost, hanem látszódnak rajta legalább az autópályák, a hidak stb.) Jogosan merül fel az igény ezen zsúfolt részek (és így természetesen a gráf egy tetszőleges részének) kinagyítása iránt, hogy a gráf részleteiben is vizsgálhatóvá váljék.

A kinagyított kép szintén legyen mérethelyes, ugyanúgy lehessen rajta dolgozni (például pontot keresni, élt törölni stb.) mint a kezdőképen. Természetesen nagyítási funkciót ezen a már kinagyított képen (aktuális kép) is biztosítani kell, hiszen lehet, hogy egyetlen nagyítás nem elegendő a hálózat egy adott részletének kényelmes vizsgálatához. (Például ha a főváros utcáit szeretnénk megnézni egy országos úttérképen.)

Ha van nagyítási funkció, akkor biztosítanunk kell a visszalépés lehetőségét is, azaz ha az összes „előző” kép helyreállítását nem is, de legalább a kezdőképre való visszalépést igen. Onnan ugyanis újbóli nagyítással/nagyításokkal a kívánt részletre „pozicionálhatunk”.

Az adatmódosítások eredménye/hatása jelenjen meg a képernyőn. Elsősorban persze az eredeti/fizikai koordinátahatárok módosítására gondolunk, hiszen ekkor a képernyőre való transzformálás paraméterei is megváltoznak. (Gondoljunk csak egy, az eddigi pontoktól viszonylag távol lévő pont felvételére.)

Végezetül még egy utolsót. A gyakorlati alkalmazásokkal ellentétben — ahol általában sok ponttal dolgoznak, s így nincs lehetőség a pontok szemléletes kirajzolására (például adott sugarú körökkel), hiszen egymást is, meg a köztük lévő éleket is „eltakarnák”, itt, a mellékelt példaprogramokban (ahol is max. 26 db pontunk lehet összesen) lehetőségünk nyílik akár többféle pontméret választására is. Természetesen csak akkora pontméret lehet egy adott képen, amekkora még nem okozza a pontok és élek takarását. Az aktuális pontméretet, a koordinátákat módosító eljárások (új pont felvétele, egy meglévő pont más helyre mozgatása stb.) is befolyásolják, így ezek után a pontméretet is aktualizálnunk kell.

Vizsgáljuk meg ezeket a főbb tevékenységeket szép sorjában. A rövidség és egzaktság érdekében bevezetjük az alábbi, talán mindenki számára érthető Pascal nyelvű deklarációkat:

```
Type KoordInt = Integer;
Koord = -999999..999999;
FizKeret = Record
MinX, MaxX, MinY, MaxY :Koord;
End;
KepKeret = Record
MinX, MaxX, MinY, MaxY :KoordInt;
End;
Var Adat :FizKeret; { Az
eredeti/fizikai koordinátahatárok }
RT :KepKeret; { A rajzterület
határai a képernyőn }
```

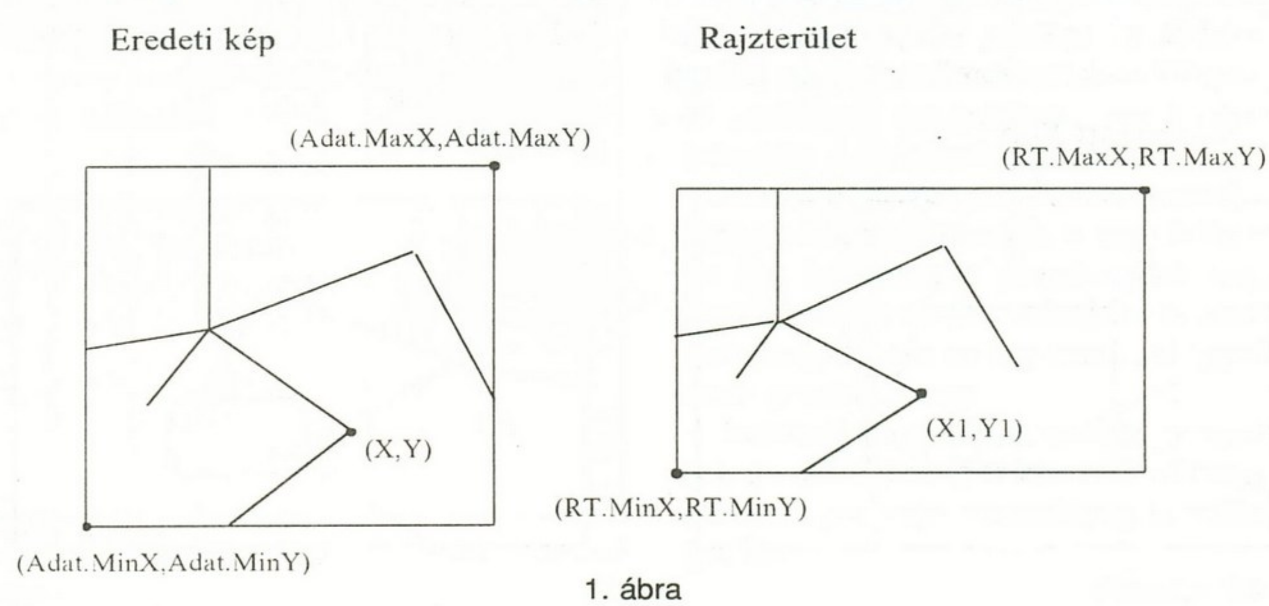
Kezdőkép

Hogyan transzformáljuk az eredeti/fizikai koordinátákat a rajzterületre? (1. ábra) Legyen:

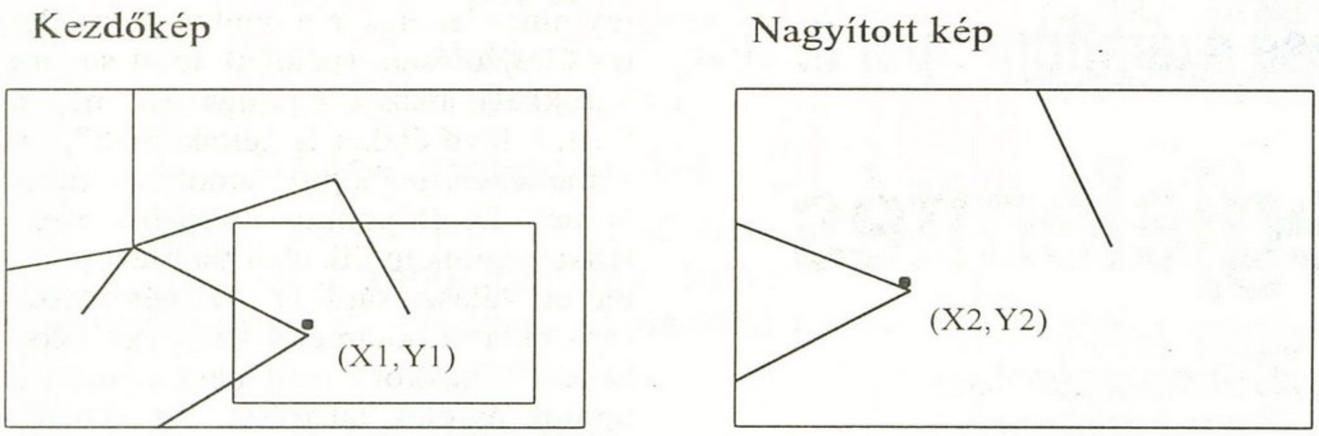
```
XSzorzo = (RT.MaxX – RT.MinX) /
(Adat.MaxX – Adat.MinX) és
YSzorzo = (RT.MaxY – RT.MinY) /
(Adat.MaxY – Adat.MinY) ill.
Szorzo = Min(Xszorzo, YSzorzo).
```

Ezután egy adott (X,Y) pontra elvégzett

```
X1 = Szorzo * (X – Adat.MinX)
Y1 = Szorzo * (Y – Adat.MinY)
```



1. ábra



2. ábra

transzformációval a (0,0) és az (RT.MaxX — RT.MinX, RT.MaxY — RT.MinY) pontok által meghatározott téglalapba eső (X1,Y1) pontot kapunk (az (X,Y) pont „képét"), amelyet már kirajzolhatunk a rajzterületre.

Nagyítás

Szükségünk van egy nagyítókeretre, amellyel a rajzterület egy tetszőleges része kijelölhető. Azaz tudnunk kell a rajzterület arányait megtartó téglalapot kirajzolni, méretezni és mozgatni. A kirajzolás nem probléma, az egyedüli feladatot a keret alatti terület elmentése és visszaállítása jelenti, hiszen a keret kirajzolásával nem „ronthatjuk" el a képünket. Ezt egy egydimenziós tömb segítségével meg is oldhatjuk, ha tároljuk benne a nagyítókeret által éppen eltakart pontok koordinátáit és színeit (2. ábra).

```
Var NK :KepKeret; { a nagyítókeret
helyzete a kezdőképen }
```

Ha tehát tudjuk nagyítókeretünk helyzetét, akkor a nagyítási szorzó meghatározható az

```
NSzorzo = (RT.MaxX - RT.MinX) /
(NK.MaxX - NK.MinX)
```

aránnyal. Ezután már az

```
X2 = NSzorzo * (X1 - NK.MinX)
Y2 = NSzorzo * (Y1 - NK.MinY)
```

transzformációval a keresett (X2,Y2) ponthoz, az eredeti (X,Y) pont kinagyított „képéhez" jutunk. Mielőtt azonban az (X2,Y2) pontot kirajzolnánk a képernyőre, a következő technikai jellegű dolgokat meg kell még tennünk (lévén, hogy Pascal grafikát használunk):

— Meg kell vizsgálni, hogy az (X2,Y2) pont a –MaxInt..MaxInt intervallumba esik-e, mert ha nem, akkor „ilyenné kell tennünk".

— Az Y2 koordinátát az Y2 = RT.MaxY–RT.MinY–Y2 képlettel „meg kell fordítanunk", hiszen az origó a képernyő bal felső sarkában van.

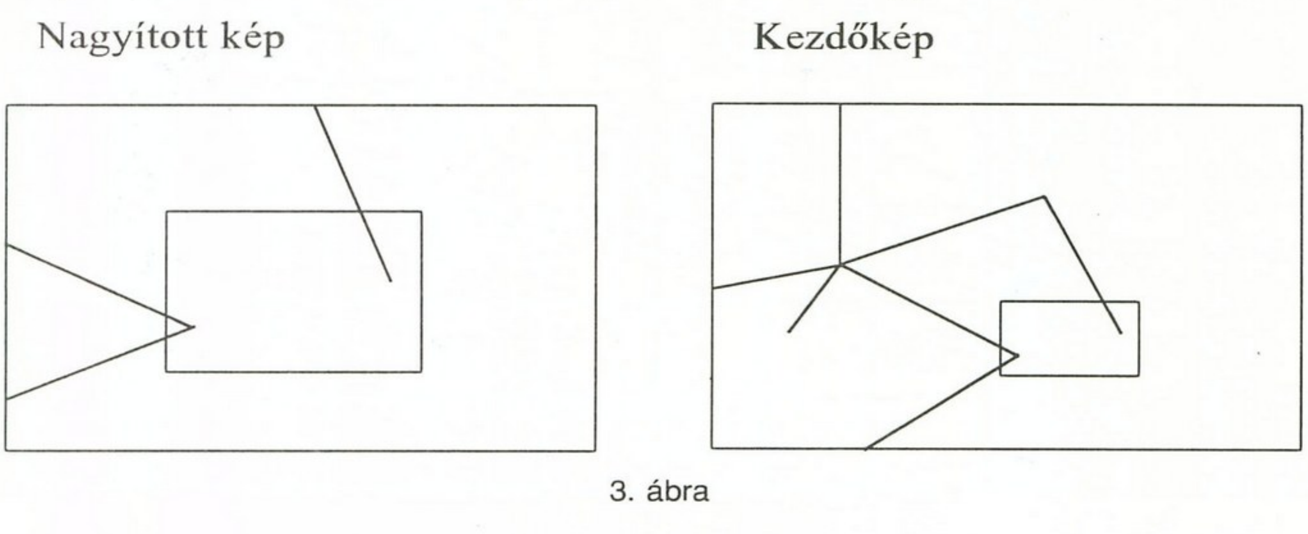
— Egy grafikus ablakot kell nyitnunk a rajzterületre, amely az abból kieső rajzrészeket levágja.

Azaz nem kell törődnünk a rajzterületről „lelógó" részekkel, ezeket a Pascal grafika automatikusan levágja helyettünk.

Nézzük meg azonban azt, hogy mi legyen akkor, ha egy már nagyított képen szeretnénk tovább nagyítani. Ezt úgy oldhatjuk meg, hogy „visszaszámoljuk" (az aktuális nagyítási szorzóval elvégzett „inverz" transzformációval), hogy az aktuális képen lévő nagyítókeret hol és mekkora lenne a kezdőképen (3. ábra).

Ezzel a „kezdőképen lévő" nagyítókerettel és az újraszámolt nagyítási szorzóval elérhetjük tehát azt, hogy újra csak a kezdőképet kelljen nagyítanunk. Persze az egész úgy hat majd, mintha a nagyított képet nagyítottuk volna tovább.

A dolog azonban nem mehet vég nélkül, amikor is a kezdőképen már túl pici lenne a nagyítókeret (például egy 2x2-es keretnél kisebb), akkor egyszerűen nem nagyítunk tovább.



3. ábra

Pontméret

Az aktuális képen használható maximális pontméret meghatározásához meg kell keresnünk a rajzterületre eső (azaz az aktuális képen lévő) pontok között lévő legkisebb távolságot, majd ennek függvényében a legnagyobb, még használható pontméretet, amelyikkel (még a két legközelebb eső pont közé is) takarás nélkül elférnek a pontok körei és egy kis hely (például sugárnyi méretű) még marad is, a köztük lévő esetleges éladatok számára (4. ábra).

Adatmódosítások

Amint azt korábban már említettük, azoknak a koordinátáknak a módosításai érdekesek, amelyekkel megváltozik az adatok fizikai határa, vagy az éppen módosított pont „lelógna" az aktuális képről.

Előbbi esetben ugyanis újra kell számolnunk a transzformáló szorzót (Szorzo), míg a másodikban célszerű a kezdőképhez visszatérni, hogy a módosítás eredményét (pl. az újonnan felvett pontot) lássuk is.

Mivel egy eredeti/fizikai koordináta aktuális képre transzformálásánál mindig áthaladunk a nagyító transzformáción is, a kezdőképhez az NSzorzo = 1 beállítás és a nagyítókeret rajzterület szélére való helyezése szükséges, mivel ekkor a nagyítás „hatástalan" marad.

Az általános szempontok után nézzük meg most konkrétan, hogy miképpen, milyen tevékenységek után lesz a tárolt adatainkból grafikus kép.

Kezdőtevékenységek

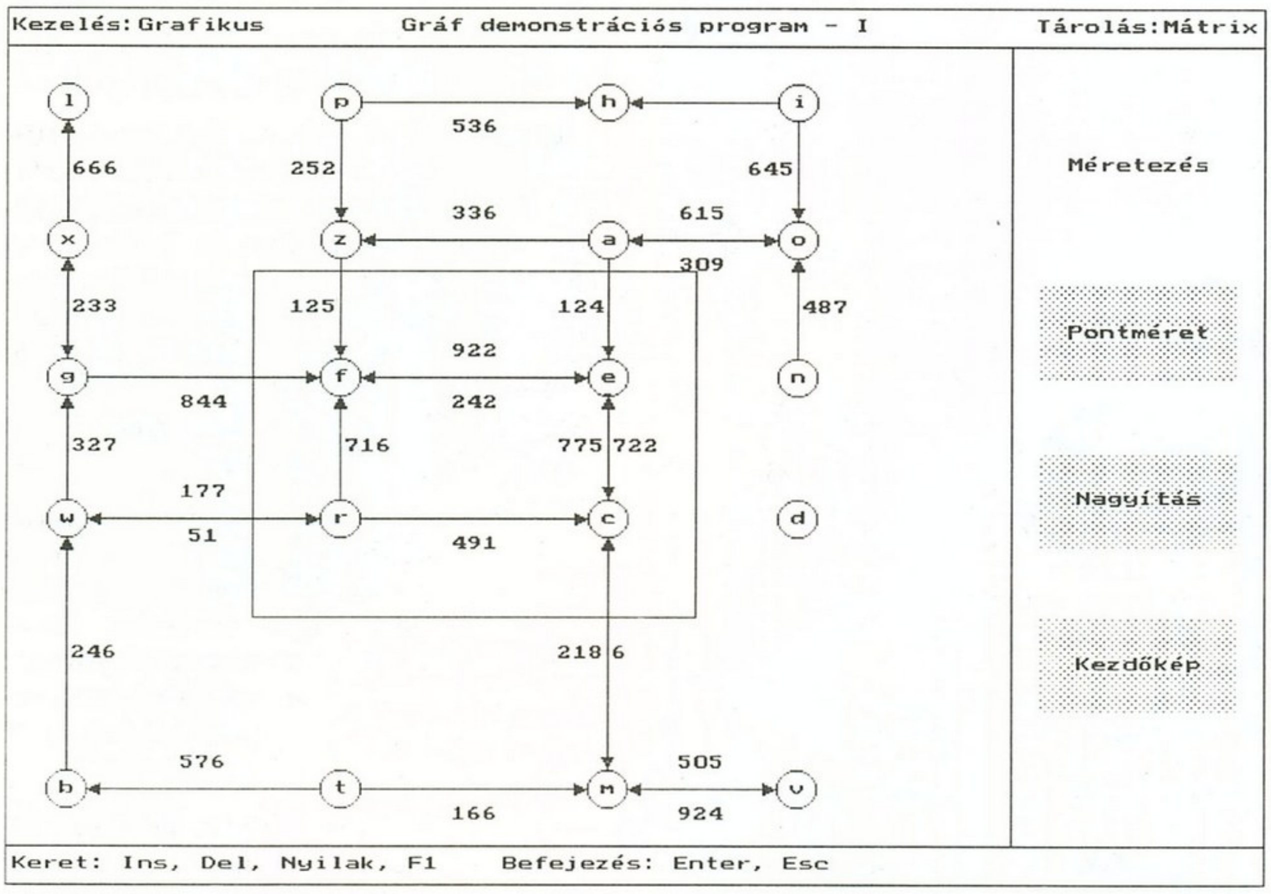
Mivel az egész munka a grafikus felületen zajlik, a legelején inicializálnunk kell a grafikus képernyőt és üzemmódot. (Az egér inicializálása is itt, a program legelején történik. Megjegyezzük azonban, hogy példaprogramjaink teljes egészében kezelhetők billentyűzettel is, és a grafikus lekérdezések is működnek egy, külön erre a célra bevezetett, grafikus „célkereszt" kurzorral.)

A program legvégén történik meg majd a grafika lezárása, a szöveges üzemmódba való visszatérés.

Meghatározzuk a koordináták határait, és beállítjuk a Szorzo értékét.

Először kezdőképet rajzolunk, tehát nem nagyítunk (NSzorzo = 1, a nagyító keret a rajzterület szélére kerül).

Beállítjuk a használható maximális pontméretet.



4. ábra

A gráf kirajzolása

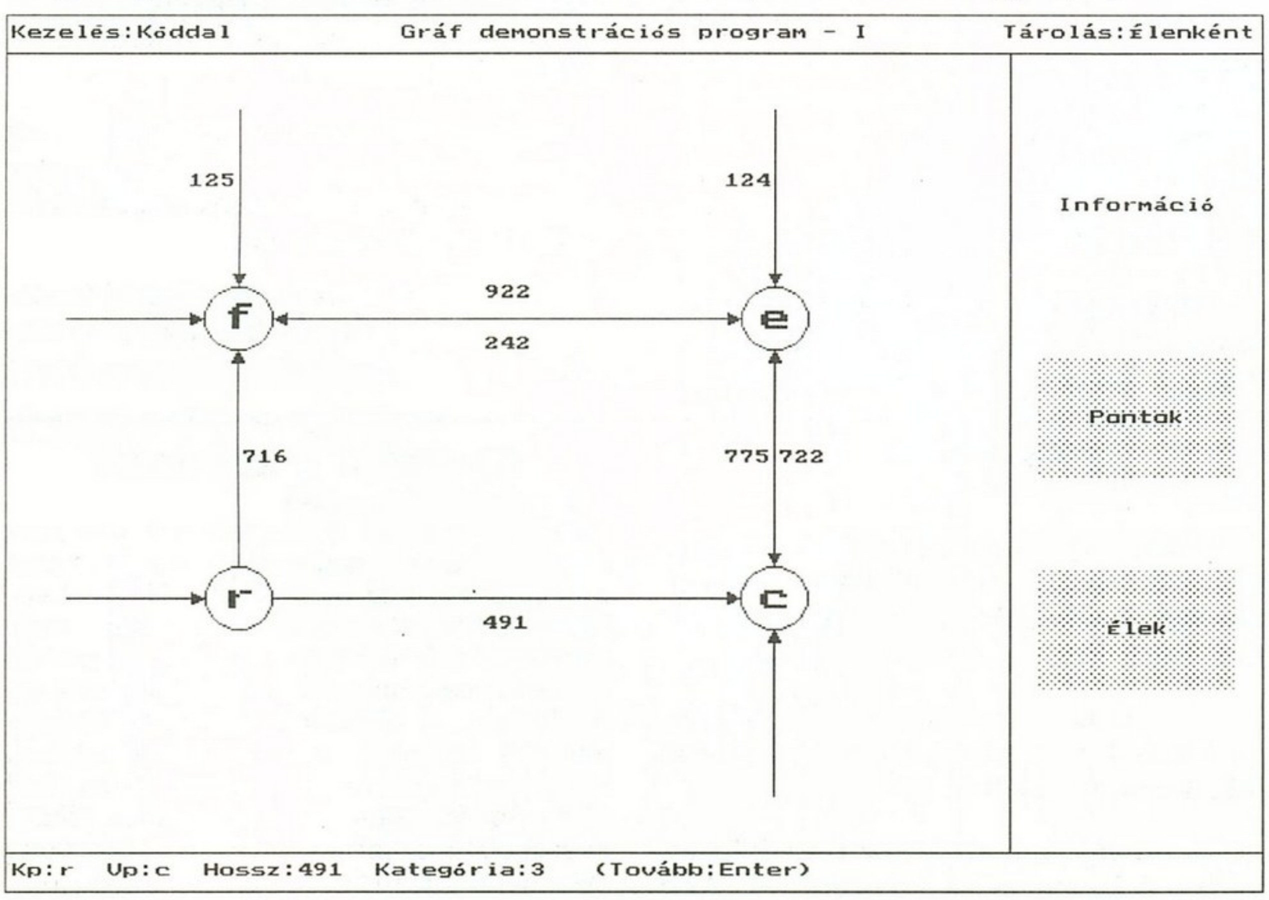
A koordinátákat mindig transzformáljuk, felhasználva az aktuális Szorzo, NSzorzo értékeket, a nagyítókeret (NK) és a rajzterület (RT) határait. Kirajzoljuk a pontokat a ponttömb alapján. (Kirajzolás előtt célszerű megvizsgálni, hogy a rajzterületre esik-e az adott pont, mert ha nem, felesleges a pont kirajzolásával vesződni, úgysem fog látszani.)

Kirajzoljuk az éleket

A kirajzolás „gyorsítása" végett itt is megvizsgálható az él rajzterületre esése, csak az a kérdés, hogy a képernyőre esés eldöntése vagy a („nem látszó" helyre való) kirajzolása „fut-e le" gyorsabban (5. ábra).

Éltárolás: végighaladunk a pontokon mint kezdőpontokon a ponttömbben, és minden pontra a belőle kiinduló éleken a mutató tömb segítségével, majd az adott két pont közé élt rajzolhatunk. (A kezdőpont ugye adott, az élvégpont pedig az éltömbbeli élrekordból nyerhető, csakúgy, mint az él többi adata: hossz, útkategória.)

Mátrixos tárolás: végighaladva az élmátrixokon, ahol élet találunk (például Hossz = 0), ott a megfelelő indexek mint pontindexek alapján, a ponttömbben tárolt koordinátákkal és az élmátrixok megfelelő adataival élt rajzolhatunk.



5. ábra

Grafikus lekérdezések

Az információk grafikus felületen történő lekérdezéséhez szükségünk lehet az egérkurzorhoz, illetve a grafikus kurzorhoz legközelebb eső pontra, élre. Az előbbi meghatározása a pontkoordináták alapján triviális, de célszerű figyelnünk az illető pont rajzterületre esését, hiszen zavaró lehet olyan pont „megtalálása", amelyik nem is látszódik a képernyőn. A legközelebbi él megkeresése, illetve annak eldöntése, hogy egy él akár csak egy kis darabja is rákerül-e a képre vagy sem, már igazi koordinátageometriai „csemege", és érdekességükön túl persze nem csak ebben a funkcióban lehetnek hasznosak. (Például: ha egy úthálózat élein lévő forgalmakat az élek vastagságával jelöljük, akkor szükségünk van az aktuális képen lévő élek maximális forgalmi adatára az élvastagság kalibrálásához.)

Biztosan előre...

Témánk végére érkezve bízunk abban, hogy ennyi kiegészítéssel példaprogramjaink tanulmányozása nem jelent majd nehézséget, és közelebb jutunk más, hasonló jellegű problémák megoldásához is. Példáinkban már megtalálhatók a képernyő-transzformációk, a „nagy" koordinátájú hálózatok generálása és megjelenítése, valamint a grafikus lekérdezések, a pont és élinformáció.

A gráfadatok módosításának lényeges grafikus vonzatairól már beszéltünk, azonban magáról a hálózat-/gráfadatok karbantartásáról még nem, erről a következő számban olvashatnak majd részletesen.

Végül ismét megadunk néhány gyakorló feladatot:

Oldjuk meg a kétirányú élek „vonalának" színezését; például az él közepétől egyik irányban az „arra" mutató él „színével" színezzünk, míg a másik irányban a másik él „színével"!

A nagyítókerettel a szélső pontok is legyenek kinagyíthatók, a nagyítókeretet úgy is lehessen pozicionálni, hogy ezek a pontok a keret belsejébe essenek, azaz nagyításkor ne legyenek „elvágva" ezen pontok körei!

Ismételt nagyítások esetén biztosítsuk az aktuális képet közvetlenül megelőző képre való visszalépés lehetőségét is!

Pusztai Pál

E SZÁMUNK HIRDETŐI

MIKROBAZÁR

A Mikrobazár rovatban a nem kereskedelmi célú egyéni hirdetések közlése ingyenes.

A kereskedelmi célú apróhirdetések tarifája gépelt soronként (azaz 60 karakterenként) 300 forint.
A terjedelem alapján így kiszámított összeget kérjük átutalni az Új Alaplap Kiadói Kft számlájára (OTP, 11701004–20171649), vagy feladni postai utalványon a kiadó címére (1539 Budapest, Pf. 571), és feltüntetni, hogy „Új Alaplap, apróhirdetés". A befizetést igazoló szelvény másolatát — a hirdetési szöveggel együtt — a szerkesztőséghez (a kiadóéval azonos címre) küldjék el.

Szerzői jogokat sértő szoftverhirdetéseket nem közlünk le.

Bármilyen típusú szöveg fordítását vállalom angolról magyarra, magyarról angol nyelvre, illetve kiadványok látványtervezését, szerkesztését is. Cím: Lachner Zoltán, 1195 Budapest XIX., Jáhn Ferenc u. 14/a. Telefon: 157-0308.

OBJECTS 2.0 — objektumorientált programozás CLIPPER-ben. Tájékoztató kérhető az alábbi címen: Szűcs János, 4400 Nyíregyháza, Vasvári Pál u. 37. Tel.: (42) 437-331 vagy 465-666/1382-es m.

Adatmentés CD-re, streamerre; winchesterről, floppyról. Ugyanitt beszerzési tanácsadást, hálózattervezést és programkészítést is vállalok. Cím: Kovács Lajos, 1031 Budapest III., Vízimolnár u. 10. IV/33.

Alaplapcsere, memória-, winchester- és floppybővítés a helyszínen. MegaSoft. Telefon: 295-5085.

Stúdiómban megbízhatóan, ellenőrzötten lefordítom angol, német, francia és magyar nyelvről/nyelvre műszaki és közgazdasági folyóiratok cikkeit, hardver- és szoftverleírásait. Áfás számlát állítok ki. Cím: Szász György, 1035 Budapest III., Kórház u. 25. Tel.: 168-4874.

Adatrögzítést, szövegszerkesztést, kiadványok, dolgozatok, évkönyvek számítógépes kidolgozását **vállalom** magyar, német, francia és angol nyelven, nyomdakész állapotig. Telefon: 135-3174 (az esti órákban).

Eladó eredeti A/3-as méretű Rotring rajztábla eredeti tartozékokkal. Irányár: 10 000 Ft. Cím: Ezüstszegi Ferenc, 1047 Budapest, Perényi Zs. u. 54. I/40.

Informatikus szakmérnök 10 éves Novell-operátori, Internet, Unix gyakorlattal **állást keres**. Telefon: 06-30-421-402.

Eladó egy SoundBlaster 16CSP hangkártya Adaptec SCSI-2 vezérlővel. Ára: 21 000 Ft. Továbbá eladó egy SANYO 4x-es SCSI CD-ROM meghajtó. Ára: 13 000 Ft. Telefon: (23)367-611.

Eladó: PC-s játékok első kötete 650 Ft; Korongba zárt gondolatok (CD-ROM) 700 Ft, Hutchinson diákenciklopédia: Fizika, Számítástechnika és Multimédia 750 Ft/db; Dedinszky Ferenc: Kézenfogva (A gépvásárlástól a programozásig, Barátom a winchester) 360 Ft/db; 1 db C64-es kazetta (90 perces) 600 Ft-ért. Cím: Kiss Balázs, 2890 Tata, Május 1. út 5. I/7. Telefon: (34)384-075.

Keresem Simonyi: A fizika kultúrtörténete; Finkelnburg: Bevezetés az atomfizikába; Bisztray: Jókedvű magyar tudósok; Károlyházy: Igaz varázslat; Hawking: Az idő rövid története; Feynman: A fizikai törvények jellege című könyveket megvételre. Cím: Győri Sándor, 1121 Bp., Mártonhegyi út 31/c. B1. I/5.

Keresem újságok, illetve könyvek lemezmellékletét, vagy annak másolatát 3,5"-os lemezen. Cím: Bencze Balázs, 2315 Szigethalom, Nyár u. 8.

SEIKOSHA OP 108-as lézernyomtató nagyon jó állapotban eladó. Érdeklődni a (24)450-514-es halásztelki telefonszámon lehet.

Vásárolnék GUS hangkártyát kedvező áron. Cím: Szarka Endre, 8500 Pápa, Fő u. 24. Telefon: (89)310-110.

Eladó Sinclair Spectrum (48 K) számítógép, tartozékokkal: interfész I. + microdrive, joystick interfész + 2 db joystick, SEIKOSHA GP 50S típusú nyomtató, a teljes magyar nyelvű irodalom, rengeteg program és egyéb extrák. Irányár: 25 000 Ft. Telefon: 149-6265.

Nagyon olcsón eladók: régi, 386-os gép, valamint EPSON FX1000, Fujitsu DX 2200 (soros portról üzemel) és Commodore MPS 1000 (C64-gyel és PC-vel is használható) nyomtatók . Telefon: 275-9550.

Mintegy 400 darabból álló (megkímélt állapotú) rock, beat, jazz, dixi, country stb. **hanglemezgyűjtemény eladó**. Telefon: 215-9224 (egész nap).

Keresem az Új Alaplap régebbi kiadásait, lemezeit 1990-től. Telefon: (79)356-667.

Elnézést a késésért

Az Új Alaplap a megjelenési ütemterv szerint minden hónap első 10 napjában eljut a hírlapárusokhoz és az előfizetőkhöz. Legutóbbi számunk eleve késői, április 8-i megjelenésre volt tervezve, és ezt a gyártás végső fázisa (ahol a mellékletek felszerelése, a fóliázás, a címkézés stb. történik) egy váratlan technikai akadály következtében még megtoldotta néhány nappal. Ezért jutott el az áprilisi szám sok olvasónkhoz jókora késéssel. Elnézést kérünk, és reméljük, hogy ilyesmi nem sűrűn fog megismétlődni.

Számítástechnika és a vakrepülés

Kérdésekre váró Válaszok

Ez a könyvismertetés szervesen kapcsolódik a hónap témája összeállításhoz. A „Válaszok" könyvsorozat ugyanis a konkrét tárgyon messze túlmutató gondolatokat ébreszthet az Új Alaplap olvasóiban. Ezért is indul írásunk egy kicsit hosszabb kerülővel.

1996-ban súlyos légikatasztrófák voltak. Sokszor feltették a kérdést a szakembereknek, hogy mivel magyarázható ez, hiszen inkább a balesetek számának csökkenését lehetne elvárni: a repülést korszerű technika szolgálja ki, a fedélzeti számítógépek automatikusan és megbízhatóan megoldanak egyre több olyan feladatot, amelyet korábban emberi beavatkozásra kellett bízni.

Vészkijárat az automatából

A polgári repülés legújabb szerencsétlenségei mögött azonban érdekes összefüggésekre lehet felfigyelni. Gyorsan nő a légiforgalom, igen erős az árverseny. A verseny által leszorított árú repülőjegyek miatt pedig nem mindenütt keletkezik akkora bevétel, hogy lehetővé váljon a géppark gyakori cseréje, viszont az öreg gépek megfelelő karbantartása és felújítása szintén sokba kerül. Ugyanígy költséges a pilóták magasabb szintű felkészítése. A légitársaságok szinte minden területen rákényszerülnek a „spórolásra", amennyire a nemzetközi légiforgalom biztonsági előírásai ezt egyáltalán még elviselik. A hibák bekövetkezésének valószínűsége szükségképpen nő, jóllehet statisztikailag nézve a repülőgép továbbra is kiemelkedően a legbiztonságosabb közlekedési eszköz.

Az elöregedő géppark (286, 386...), a felújítás (upgrade) nehézsége, a meghibásodás gyakorisága (fatal error) ismerős a számítástechnika világában is. Azonos gazdasági körülmények többnyire hasonló következményekkel járnak minden ágazatban. Ezekkel nem a mi lapunk feladata foglalkozni, de van a légikatasztrófák okai között egy olyan is, amely erősen számítástechnikai vonatkozású — közvetlenül is, átvitt értelemben pedig még inkább.

Egyes légitársaságok korszerű gépeinek műszaki hibái azért vezettek tragikus végkifejlethez, mert a pilóták túlságosan rá voltak bízva a számítógépekre, és a kritikus pillanatokban képtelenek voltak átvenni a vezérlést a túlautomatizált rendszerektől. Márpedig azokban a vészhelyzetekben, amikor nagyon bonyolult, nagyon nehezen áttekinthető környezetben fellépő hibákat kell másodpercek alatt kiküszöbölni, sokszor nincs más megoldás, mint kiiktatni a megzavarodó, és abban a pillanatban már nem „mindentudó" elektronikus rendszereket, vagyis kézi irányítással, a régi repülési technikával kell manőverezni. Ehhez két dolog kell. Egyrészt: a legkorszerűbb gépekkel (is) lehessen repülni elektronika nélkül, másrészt: a pilóták legyenek kiképezve a hagyományos repülésre (is). Ezzel sok gép és emberélet lenne megmenthető. A majdnem bekövetkezett balesetek nagy részében a „szerencse" fő forrása többnyire éppen az volt, hogy a pilóta fizikailag, manuálisan uralni tudta a repülő szerkezetet, a „megtöbbszörözött, szuperbiztos" elektronikára épített vakröptető „leváltásával", kiiktatásával.

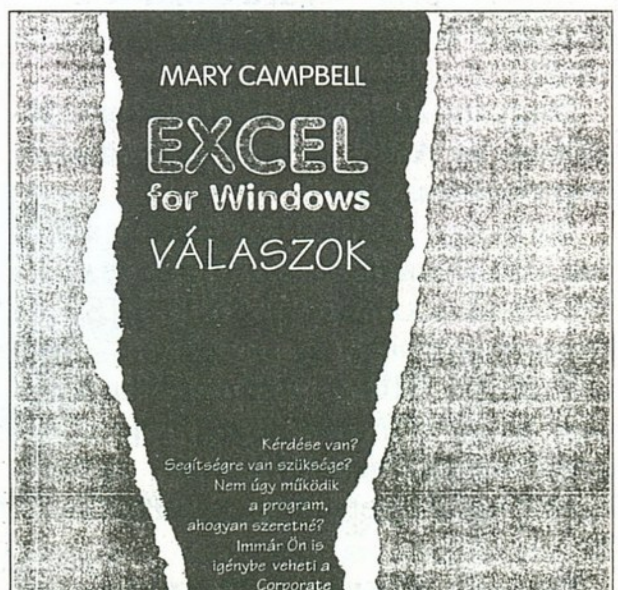
Még nem „fekete doboz"

Az iménti gondolatok a Panem kiadó gondozásában, a Válaszok sorozatban megjelent könyvek olvasása közben vetődtek fel bennem. E kitűnő kézikönyvek mindazok könyvespolcán jó helyen vannak, akik a számítógépet nem „fekete dobozként" kezelik, hanem tudni akarják, mi miért és hogyan történik benne, és ha valami nem stimmel, azt maguk szeretnék megérteni és rendbehozni. A Mikroszámítógép Magazinnak majd az [Új] Alaplapnak is kedvelt írásai voltak azok, amelyek tulajdonképpen az olvasóktól beérkezett szakmai kérdésekre válaszoltak, vagy „tippeket és trükköket" adtak — többnyire olyan problémákra, amelyek jobban átgondolt operációs rendszerek és lelkiismeretesen kimunkált programok esetén fel sem merültek volna.

Sokkal nagyobb azonban a számítógépet fekete dobozként használni akarók tábora, és meg is indult az a fejlesztés, amelynek révén a számítógép a tévével, a mosógéppel és a gépkocsival közel azonos használati jellegű készülékké válhat. Most azonban itt még nem tartunk. Képzeljük el, milyenek lennének az autók, ha a személyi számítógépek és a programok mai fejlettségi színvonalán gördülnének le a szerelőszalagról! Például a Fenestra típusú gépkocsi idei változatának piacra kerülésekor több száz oldalas könyvnek kellene részleteznie a kocsi használata során fellépő hibák orvoslására vagy áthidalására ajánlott tennivalókat, kezdve attól, hogy miért nem tudjuk elindítani, folytatva az útburkolat iránti érzékenységén át egészen a 100 km/h alatti sebességnél fellépő vibrálásig.

A számítástechnikát mélyebben értőknek persze „kutya kötelességük" az utolsó driverig minden részlettel megismerkedni, és ha egy „kész" termékben 1500 hiba van, akkor is elsajátítani a korrigálás módját (ha már a „gyárban" ezt nem tudták vagy nem voltak hajlandók megcsinálni).

Csakhogy a számítógép általános használati eszközként valahol félúton helyezkedik el a „bitvadászok lőtere" és a „bolondbiztos fekete doboz" között, s egyre több jel utal arra, hogy ebből a felemás pozícióból nem is nagyon akaródzik neki elmozdulnia, jóllehet erre az utóbbi időben egyre többen noszogatják.



Egy repülőgép szakmai dokumentációja kinyomtatva a repülőgép súlyával azonos mennyiségű irdatlan papírtömeg is lehet. Mindebből az utasnak, aki felül a gépre, gyakorlatilag semmit nem kell tudnia. A pilótának már viszonylag részletes ismeretekkel kell rendelkeznie, de az övé is elég távol áll a teljes tudástól. Ezzel szemben a számítógépet nem lehet „egyszerű utasként” rendeltetésszerűen birtokba venni, még ha elég tekintélyes is azoknak a száma, akik „vakrepülésben” használják. (Hasonlítsuk csak össze az országban működő számítógépek számát a számítástechnikai szaklapok és szakkönyvek példányszámával!) Viszont azok, akiket egyáltalán érdekel, hogy működés közben miket üzenget a számítógép, akarva-akaratlanul olyan problémákba botlanak, amelyekre a válasz sokszor a programrendszerek (és a hardverösszefüggések) legrejtettebb bugyraiból kaparható csak elő. Összeválogattunk néhány (terjedelmi okokból csak rövid) példát a „DOS válaszok” című könyvből:

Amikor elindítottam a számítógépemet, az „Unrecognized command in CONFIG.SYS line” hibaüzenet és egy sorszám jelent meg a képernyőn. Hogyan tudom kiküszöbölni ezt a hibát?

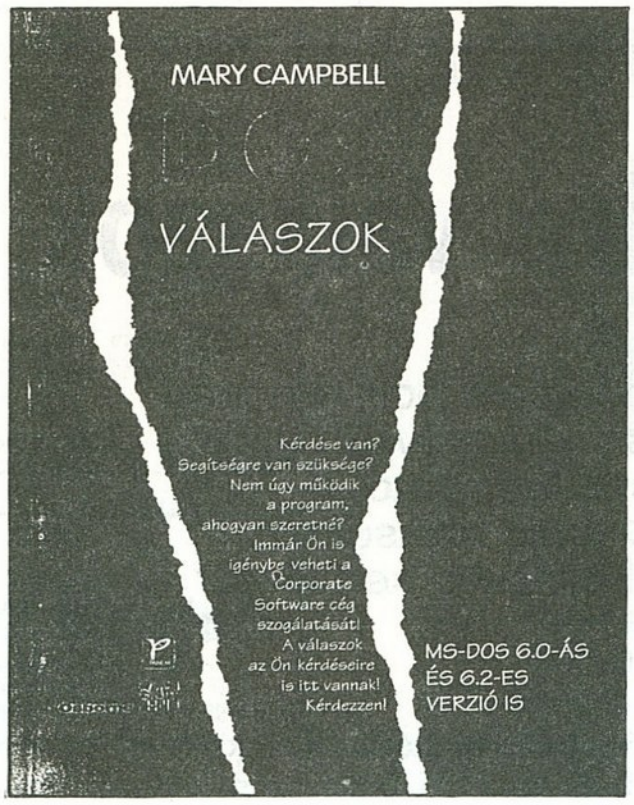
Az a sor, amelyet a DOS nem tudott értelmezni, valószínűleg hibásan beírt parancsnevet vagy egy olyan állomány nevét tartalmazza, amely nem az előírt alkönyvtárban van. Indítsa újra a számítógépet, és amikor megjelenik a „Starting MS-DOS...” üzenet, nyomja le az *F8* billentyűt, majd a sorokat egyesével végrehajtva — valamennyi kérdésre **Y**, azaz igen választ adva — keresse meg azt a sort, amelyik a hibaüzenetet okozta! Végül a CONFIG.SYS állomány átszerkesztésével javítsa ki a hibát!

Hogyan lehet escape karaktert előállítani az ANSI escape szekvenciák végrehajtásához?

Az escape karakterre általában akkor van szükség, amikor ANSI escape szekvenciát akar küldeni az ANSI.SYS eszközmeghajtóhoz. Ha az escape szekvenciákat a PROMPT parancsban használja, akkor pl. megváltoztathatja a képernyő színét. Escape karaktert a következőképpen lehet előállítani: PROMPT $e

Hogyan alakulnak ki az elveszett allokációs egységek (lost allocation units) és a keresztbe csatolt klaszterek (cross-linked clusters)?

Az elveszett allokációs egységek és a keresztbe csatolt klaszterek annak



eredményeként alakulnak ki, hogy az operációs rendszer elveszti az információ nyomát a hajlékony- vagy a merevlemezen. Az ilyen jellegű problémák kialakulásának számos oka van. Pl.:

— áramkimaradás vagy áramingadozás a számítógép energiaellátásában;

— hibásan működő meghajtó vagy meghajtóvezérlő;

— a rendszer újratöltése, mialatt egy program éppen a lemezre ír;

— feladatmegosztásos rendszerben dolgozva egyszerre több program ír a lemez azonos területére;

— az egyik program nem működik megfelelően;

— az egyik program nem képes együttműködni a tömörített lemezzel;

— egyéb hardver- vagy szoftvergondok.

A számítástechnika épülete

Jó tehát, hogy vannak ilyen könyvek, bár egyesek szeretnék elhitetni velünk, hogy ez a tudás egyre inkább felesleges, és „a program” mindent elvégez helyettünk, nekünk csak az ikonokra és a menüpontokra kell kattintgatnunk. „A mi pilótafülkénk” azonban elég keserves tapasztalatokkal szolgál ezekről a „szuperbiztos” automatákról. A DOS 6.0 és 6.2 verzióival kapcsolatos kérdések feszegetése idejétmúltnak tűnhet, mondván, hogy sorra jönnek az új, az „ilyen szuper, olyan szuper, automatikus, grafikus, multiez, multiaz...” rendszerek.

Egyelőre azonban maradjunk csak a realitásoknál. Ennek a könyvsorozatnak a születési körülményei nagyon is árulkodóak. A hibaüzenetekre, a programok nem mindig könnyen értelmezhető megnyilvánulásaira adott világos, tömör válaszok mögött sokkal komolyabb háttér húzódik meg, mint ahogy azt felületesen gondolnánk.

Az Osborne/McGraw-Hill könyvkiadó és a Corporate Software Incorporated (CSI) közös munkacsoportot hozott létre a felhasználóktól beérkező szoftveres kérdések megválaszolására. A CSI-nél dolgozó szakembergárda havonta mintegy 200 000 (!) telefonhívást kap. Ez a szoftveres információs szolgálat a jelentősebb szoftverfejlesztőkkel kiépített kapcsolat révén mindig „képben van”, és hatalmas adatbázisára támaszkodva tanácsot tud adni a hozzá fordulóknak. Mellesleg pedig anyagot szolgáltat a Válaszok könyvsorozat köteteinek összeállításához is. A magyar nyelvű kiadás gondozója, a Panem Kiadó külön szolgáltatást is nyújt: aki a könyvből nem kap magyarázatot a témával kapcsolatos valamelyik problémájára, az megírhatja kérdését a kiadónak, és levélben kap rá választ. Példamutató kezdeményezés!

A számítástechnika (sajnos) nem repülőgép. Nem olyan eszközrendszer, amelynek minden részlete gondosan össze van hangolva az összes többi elemmel, hanem sokak által sokféle anyagból és változatos módszerekkel létrehozott „épület”. Aki annak csak egyik szobájába költözik be, és mindig ugyanazt a kaput és lépcsőházat használja, az talán el is hiheti, hogy ebben a házban minden a legnagyobb rendben van. De ha tapasztalatokat akar szerezni az épület más részein is, hamarosan szidni fogja az építészt — és sürgősen beszerez néhány olyan könyvet, amilyenek például a Válaszok sorozatban jelentek meg.

Faklen Pál

A Válaszok sorozat kötetei:

Mary Campbell:
DOS
336 oldal, 1995

Mary Campbell:
Word for Windows 6.
472 oldal, 1995

Mary Campbell:
Excel for Windows
402 oldal, 1995

Kathy Ivens:
OS/2 Warp
282 oldal, 1996

James Nadler – Don Guarnieri:
NetWare
203 oldal, 1996

Kiadó: Panem–McGraw-Hill
Mindegyik fenti kötet ára: 1490 Ft
(A három utóbbi könyvre 30% diákkedvezmény)





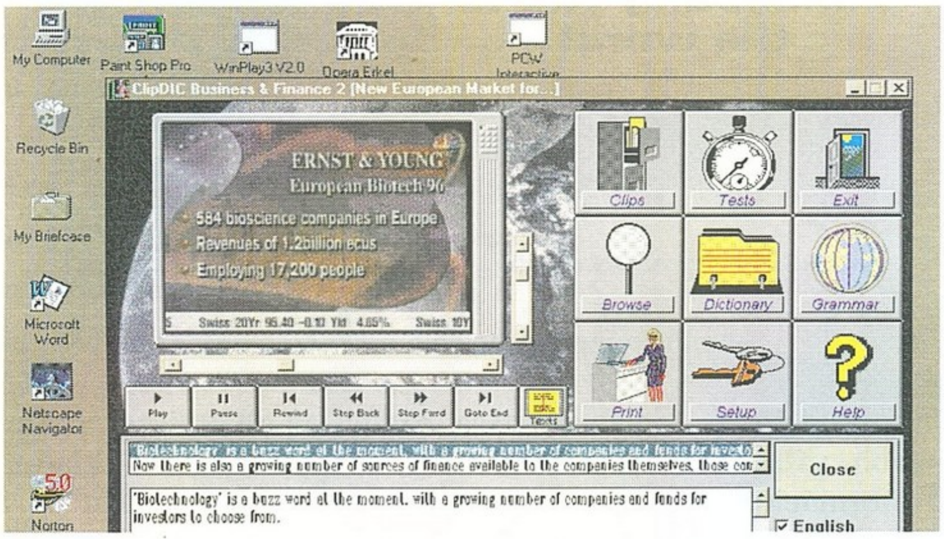
INFORMÁCIÓKÉRÉS: 0502 ▲



INFORMÁCIÓKÉRÉS: 0562 ▲

Profi munka

Már úgyszólván teljesnek tűnt a bajai Profi-Média nyelvoktató CD-inek palettája, de a szintenkénti és szakmai profilírozás a jelek szerint további finomításokat is megenged. Így a Clipdic sorozat Üzleti angol (Business & Finance)



második CD-je sem az első ilyen CD-nek a folytatása, kiteljesítése, hanem teljes értékű, önálló alkotás. Jól jelzi ezt a CD-n felölelt szakterület: Business and Technology, Emerging Markets, ERM and EMU, Marketing and Advertising, Foreign Trade and Currencies. Ezek egyikét sem tartalmazza az üzleti angolt gyakoroltató első multimédia nyelvkönyv. Kiforrott a technológia, kiváló alapanyag (például CNN, NBC), a kor színvonalán álló kép- és hangeffektusok, jól áttekinthető szerkezet, a Windows felület összes előnyét ötvöző kiszolgálás a felhasználó számára. E bevált tematika és technológia mellett a bajaiak minden évben kirukkolnak valami meglepő, igazi újdonsággal is. Idén őszre várható (éppen a Compfairre érkezik meg) az a beszédanalízissel, Internet-kapcsolattal „megfejelt" és rajzfilmekkel nyelvet oktató CD, amelynek alapanyaga az egykor igen népszerű Czobor—Horlai nyelvkönyv.

Allegris

A fenti szoftver neve nyilván sokak számára ismeretlenül cseng, csakúgy, mint a gyártó Intersolvé. A szűkebb szakmai körökben viszont igen nagy megbecsülésnek örvendenek a cég olcsónak koránt sem nevezhető fejlesztőrendszerei. Ezek időtállóságáról elég annyit mondani, hogy egyik törzsszerzőnk az akkor még Politron névre hallgató gyártó 1988-ban készült DOS-os rendszerét használja ma is, és állítása szerint e rendszer segítségével bármelyik korábbi projektje ma is gond nélkül „életre lehelhető".

Az új Allegris eszközcsalád viszont az első komponensalapú fejlesztőeszköz a piacon, amely egyszerre célozza meg a vállalati kliens/szerver és az intranet/Internet alkalmazásfejlesztést. A drag and drop alkalmazásparticionálás, a megosztott komponensek beépített támogatása, az alkalmazás-



A PHILIPS bemutatja nagy képernyőjű, kis helyigényű LCD monitorát

A Philips bemutatta kis helyigényű, karcsú, nagyteljesítményű és kisugárzásmentes Brilliance 4500AX típusjelzésű, 14,5 col átmérőjű LCD-panel monitorát.

A kis alapterület-igényű, karcsú monitorok különösen alkalmasak olyan irodai környezetben, ahol szűk a hely, valamint az olyan ipari vagy laboratóriumi alkalmazásokhoz, ahol a kisugárzás vagy az elektronikus emisszió zavarhatja a nagy érzékenységű berendezéseket. Japánban, az Egyesült Államokban és Nyugat-Európában egyre növekszik a kereslet a lapos LCD monitorok iránt, így részesedésük a monitorok piacán két évtizeden belül várhatóan eléri a 20%-ot.

A homloksíktól a hátlap síkjáig mindössze 6.3 cm vastag, az állvánnyal együtt mindössze 11.5 cm talpszélességű Brilliance 4500AX típusú színes LCD monitorok kis alapterületen, nagyméretű képet nyújtanak. A képernyő hasznos területe a 17 col átmérőjű katódsugárcsöves modellekéhez hasonló, de a Brilliance LCD-panel monitorok helyigénye azoknak mindössze egyhetede.

A Philips mérnökei úgy csökkentették az alapterületet, hogy ezért nem kellett feláldozni a képminőséget. A Brilliance 4500AX továbbfejlesztett aktív mátrixos, vékonyréteg-tranzisztoros (TFT) folyékony kristályos kijelzője a ma elérhető legélénkebb színeket, éles kontrasztos képet és a legalacsonyabb teljesítmény-felvételt kínálja. A kristálytiszta, szemnek kellemes képet 260000 színben jeleníti meg. A pontméret 1024x768 felbontásnál 0.288 mm. Az 1024x768 képpontos üzemmód 75 Hz-es képfrissítési frekvenciája szavatolja a rezgésmentes megjelenítést.

A fényerő meghaladja a 200nit értéket. A Philips cégnél kifejlesztett tökéletes fókuszálású LCD technológia kiszűri a konvergencia-hibát, és megszünteti a párna- vagy hordó-torzítást is. A monitor 640x480, 800x600, valamint 1024x768 felbontású üzemmódban működik.

A Brilliance 4500AX típusú színes LCD monitorok kompatibilisek a VGA, SVGA, VESA és MacIntosh szabványos időzítésekkel, valamint biztosítják a duplex adatátvitelt a PC és a monitor között a Windows 95 (VESA DDCI/2b) vezérlése alatt. A szabványos video-kártyákkal 15-érintkezős D csatlakozó biztosítja a teljes körű kompatibilitást.

Multimédia és video-konferencia alkalmazásokhoz a Brilliance 4500AX jó minőségű, beépített, 1W teljesítményű, előre irányított hangszórókkal rendelkezik.

A Brilliance 4500AX monitorok teljesítmény-felvétele rendkívül alacsony, maximum 33W.

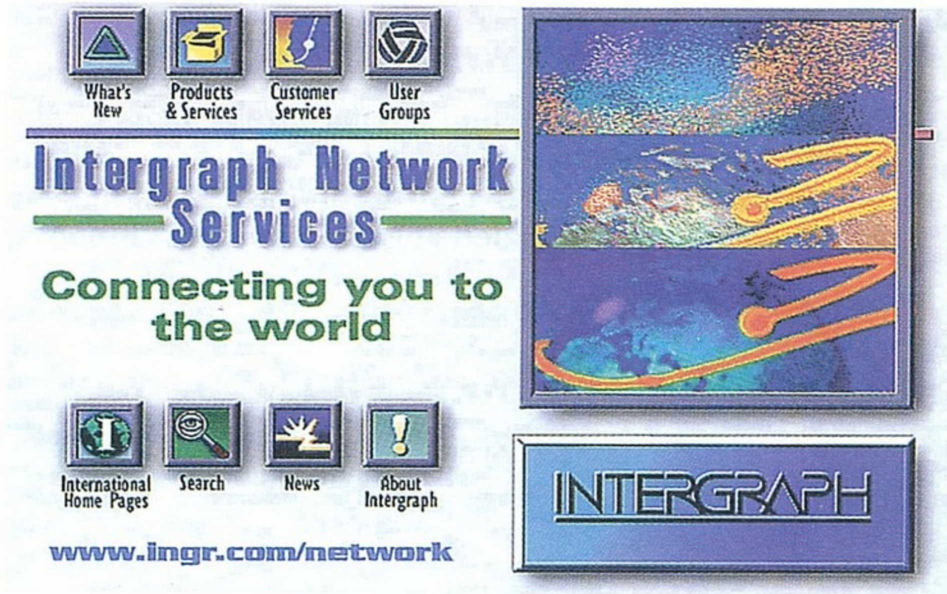
PHILIPS

INFORMÁCIÓKÉRÉS: 0542 ▶

menedzsment és a szoftverelemek vállalati szintű újrafelhasználhatósága révén az Allegris lehetővé teszi alkalmazáskritikus rendszerek igen gyors fejlesztését és használatba vételét anélkül, hogy ez magas fejlesztési és működtetési költségeket jelentene a vállalat számára.

Az Allegris eszközrendszere biztosítja a skálázhatóságot a csoportméret, az alkalmazhatóság és a technológia területén, az egyedi fejlesztéstől a (nagy)vállalati alkalmazásfejlesztésig, a kliens/szerver struktúrától a webes implementálásig. A vizuális fejlesztés egyszerűségét kombinálja a gyors komponensalapú fejlesztés robusztus képességeivel, s „mellesleg" C++-szintű hordozhatóságot biztosít a fejlesztők számára, anélkül, hogy a C++ professzionális szintű ismeretét megkövetelné.

Az Allegris rendszer alkalmazásba vétele három olyan hazai nagyvállalatnál is folyamatban van — MOL, MÁV, Budapest Bank —, amelyek tapasztalatai vélhetően irányadóak lehetnek mások számára is, és elősegíthetik szélesebb mértékű magyarországi elterjedését is. Az Allegris igazi high-end eszköz, ára is ennek megfelelő, magyarországi disztribútor a Megatrend Kft.



Az Intergraph weboldala az Interneten...

Videokonferencia

Számítógépbe építhető új videokonferencia-rendszerrel jelent meg a közelmúltban a Vidcom Kft. A készlet egy kisméretű kamerát, egy integrált videokódoló és dekódoló ISDN kártyát, egy desktop-video szoftvert, valamint mikrofont és fülhallgatót tartalmaz. Legalább Pentium processzor ajánlatos hozzá. A két változat közül a CLI Desktop Video 1000-es jelű, amely maximálisan 128 kbit/s sávszélességen képes kommunikálni, míg a 2000-es jelű 384 kbit/s-on. A rendszer szoftverkomponense a kép- és hangmegjelenítésen túl lehetőséget nyújt szokásos állóképformátumok (BMP, TIFF, JPEG) elmentéséhez, valamint közös ábrák, rajzok elkészítéséhez is. A rendszer a szabványos QCIF (176x144-es) felbontásban 30, FCIF-ben (352x288) 24 képváltásra képes másodpercenként. Windows 3.1 és Win95 alól is elindul, s támogatja például az Intel Proshare Premier, a Microsoft NetMeeting és a Databeam Farsite alkalmazásmegosztó szoftvereket. A videokonferencia ISDN vonalon, helyi hálózaton és ATM-en egyaránt megvalósítható.

A Solid Edge új változata

Sikeres nyitott hónapot — és negyedévet — zárt márciusban az Intergraph Magyarország. A konzultációsorozat és a különböző szintű bemutatók fogadtatását mindennél jobban jelzi, hogy az összes bemutató célú gépkonfiguráció „lábon elkelt" — és egyáltalán nem kiárusítási áron. Így azután az újságíróknak prezentált bemutató már „kölcsöngépeken" zajlott le.

Lapunk korábban is foglalkozott a Windows-alapú gépészeti tervezőrendszer előző változatával. A mostaniban bővült a szereléstervezés és az alaksajátosság-alapú testmodellezés funkcionalitása, ezáltal az alkatrészek szélesebb választéka tervezhető meg, mégpedig — a kiadott információk szerint — teljesítménycsökkenés nélkül és a könnyű kezelhetőség megőrzése mellett. Az új változat egyik lényeges ismérve a 2D és 3D újszerű ötvözése: a teljesen új kétdimenziós rajzolási parancsok alkalmazásával ez a rendszer támogatja a „nulláról" történő műszaki rajzolást, méretezést és gépészeti kiegészítéseket, s a tervezőnek megvan a lehetősége, hogy a háromdimenziós modellből készített nézeteket, metszeteket kiegészítse 2D rajzelemekkel. Még a 3D testmodell megváltoztatásakor is megmarad ezek geometriai kapcsolata a térbeli modellből származó rajzrészekkel.

Ezzel a megoldással és további szolgáltatások révén az Intergraphnak alighanem sikerül kockázatmentes átmeneti utat javasolni azoknak a 2D gépészeti felhasználóknak, akik fontolgatják az áttérést a 3D technológiára. Az sem elhanyagolható szempont, hogy a maga kategóriájában a Solid Edge 3.0 az első, amely megfelelt az Office 97-tel való kompatibilitás igényének. A kibővült eszköztárat kulcsrakész webtechnológia teszi teljessé: a szabványos CGM (Computer Graphics Metafile) formátumban elmentett Solid Edge-terv adatai Interneten vagy intraneten keresztül is megoszthatók, s közvetlenül megtekinthetők például a Microsoft Internet Explorer vagy a Netscape Navigator segítségével. (A Solid Edge új változatának mélyebb szakmai bemutatására következő számaink egyikében sort is kerítünk.)



...és az intraneten

A Compaqnál ez csak természetes

Compaq asztali számítógépek.

Ha olyan számítógépet keres, amelyre bizton alapozhatja üzletét, a Compaq Deskpro 2000-nél keresve sem talál jobbat. Lelke egy Pentium vagy Pentium Pro® processzor, alapkiépítésben akár 32 megabájt EDO DRAM-mal felszerelve - gyors és hatékony. 1, 1,6 vagy 2,5 gigabájtos SMART-merevlemezein bőségesen jut hely az adatok hibamentes tárolására. Része az Intelligens felügyelet - a Compaq egyedi megoldása a gondok felderítésére, mielőtt azok még bekövetkeznének, az adatok és a hardver védelmére az illetéktelen használat ellen, és egyáltalán, a PC-k felügyeletének leegyszerűsítésére.
Netán hálózati hozzáférésre van szüksége? A Deskpro 4000 és 6000 beépített nagysebességű 10Base-T Ethernet-vezérlőt tartalmaz, amely villámgyorsan 100 megabit/másodpercesre bővíthető. A továbbfejlesztett Intelligens felügyeleti funkciókból pedig nem hiányoznak a javított biztonsági riasztások és az egyszerű hálózati kapcsolódás sem.
Ha maximális teljesítményre vágyik, nos, a Deskpro 6000 igazi erőmű. Minden egyes modellben nyolcszoros sebességű CD-olvasó vagy hatszoros sebességű PC-CD-meghajtó található, a fejlett multimédiaalkalmazásokhoz szükséges audiofunkciókkal együtt.



Compaq Deskpro 2000

A legfontosabb jellemzők

Az Ön nagyobb nyugalma érdekében minden egyes Compaq Deskpro alapkiépítésben tartalmazza a hardver- és szoftvereszközök azonnali áttekintésére, frissítésére és felügyeletére szolgáló Compaq Intelligens menedzselhetőség.



Megbízhatóan támogatja a Deskpro-család a kapcsolódás különféle lehetőségeit. Minden egyes modell megfelel a Microsoft, a Novell és a többi fő hálózati operációsrendszer-gyártó előírásainak.

Legalább 5 bővítőcsatlakozót és 4 meghajtóhelyet tartalmaznak a Deskpro-család tagjai, bőséges helyet biztosítva az ön igényeinek megfelelő tárolóeszközöknek, multimédia- vagy kommunikációs kártyáknak, vagy egyéb bővítőeszközöknek.

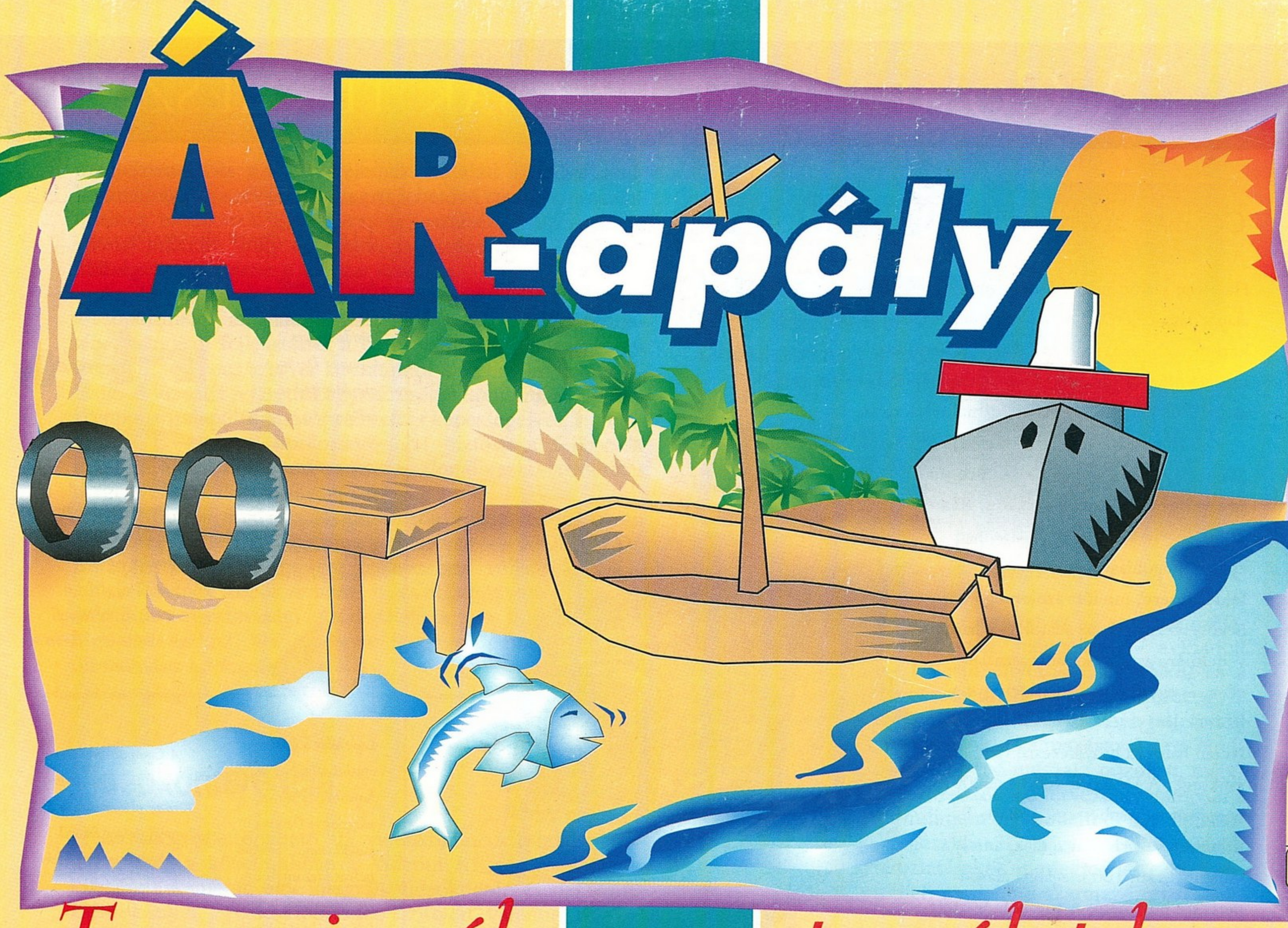
A Compaq asztali gépei az ipari szabvány előírásoknak megfelelően készülnek, és kompatibilisek az Ön cégénél használt többi PC-vel és hálózattal.

INFORMÁCIÓKÉRÉS: 0507 ▶

COMPAQ

intel inside pentium PROCESSOR

Compaq System Resellerek: Computer Lan & Wan Kft. Tel.: 188-7992 • Duna Elektronika Kft. 270-5660 • EDPland Kft. Tel.: 319-9797 • Flag Rt. Tel.: 274-2109 • Kventa Kft. Tel.: 269-5262 • Liteware Com
Kft. Tel.: 201-2936 • Mikropo Kft. Tel.: 153-0111 • Minor Rendszerház Rt. Tel.: 274-2495 • Montana Informatika Kft. Tel.: 327-9800 • Műszertechnika Computer Rt. Tel.: 260-4893 • Nádor Rendszerház Kft. Tel.:
5048 • N-Sys kft. Tel.: 173-1414 • Rolitron Informatika Kft. Tel.: 335-1323 • Sprint Computer Kft. Tel.: (30) 420-443 • Summacomp Kft. Tel.: 322-4419 • SYSTREND Kft. Tel.: 342-5145 • **Győr:** SC - Comp Kft.
(96) 318-915 • Szintézis Kft. Tel.: (96) 327-355 • **Miskolc:** Server Kft.Tel.: (46) 346-238 • **Szeged:** PC-Box kft. Tel.: (62) 486-486 • **Székesfehérvár:** Albacomp Rt. Tel.: (22) 315-414 • **Compaq Deal**
Békéscsaba: MT Békéscsaba Tel.: (66) 324-760 • Számprog Kft. Tel.: (66) 321-824 • **Budaörs:** Q-Net Kft Tel.: 326-1227 • **Budapest:** 2R Periféria Kft. Tel.: 342-3308 • Acronym Networking Ltd. Tel.: 129-6
Apostol kft. Tel.: 215-3666 • Appli-Comp Kft. Tel.: 261-5173 • Arm Kft. Tel.: 137-2620 • Compmark kft. Tel.: 270-9011 • Csada Péter Bt. Tel.: 173-2038 • Delta Elektronik Kft. Tel.: 180-1933 • DNC Tel.: 165-
• DNN Computers Tel.: 342-9903 • Donáth István ev. Tel.: 282-2524 • EcoSoft Kft. Tel.: 203-8657 • EDV-Adatszolg Kft. Tel.: 138-4043 • Első Pannon Rendszerház Informatika Tel.: 215-0766 • Fabicad Kft Tel.:
3721 • Futurecom Kft. Tel.: 212-0988 • Global Ameritech Company Kft. Tel.: 351-7327 • HR-Comp Rt. Tel.: 175-6453 • Intercas Kft. Tel.: 155-2180 • InterPC NetworkingKft. Tel.: 216-2625 • ITV Hungaroli
Tel.: 131-0851 • KFKI-Direkt Kft. Tel.: 209-3215 • Megamicro RT. Tel.: 252-1500 • Megatrend Kft. Tel.: 333-7316 • Men Bt. Tel.: 266-4653 • Mentum Kft. Tel.: 266-1767 • Módusz Rt. Tel.: 269-9006 • Net
Kft. Tel.: 363-3659 • Netx Consulting Tel.: 457-8358 • Next Computer Kft. Tel.: 185-1591 • Omikron Computer Coop. Tel.: 114-0090 • PC-Soft Kft. Tel.: 120-3442 • Plantrend kft. Tel.: 270-0838 • Porsche Inforn
Tel.: 270-5039 • Power Kft. Tel.: 252-3210 • Profi-Comp gmk Tel.: 325-9996 • Qwerty Computer Kft. Tel.: 185-2687 • Rokaat Ker. és Szolg. Kft. Tel.: 262-6302 • Rubinetti Számítástechnikai és Kereskedelmi Kft.
250-0311 • Selectrade Kft. Tel.: 294-5609 • Székely&Tsa Bt. Tel.: 410-4258 • SZÜV-Computer-M. Tel.: 163-4004 • Trigon Hardware Kft. Tel.: 285-5940 • Tronvision Bt. Tel.: 173-1095 • TZteam Kft. Tel.: 406-
• **Debrecen:** Enter Computer Tel.: (52) 311-783, • Főnixcomp Kft. Tel.: (52) 419-294 • Have Kft. Tel.: (52) 310-095 • **Dorog:** Printer Fair Kft. Tel.: (33) 442-165 • **Eger:** SZÜV-Computer-M. Tel.: (36) 313-
SZÜV Computer-M Tel.: (36) 418-430 • **Gyöngyös:** Mikrokapcsolat Kft. Tel.: (37) 313-577 • **Győr:** Comp-Sys Kft. Tel.: (96) 439-658 • SZÜV Rt., Győri Tig. Tel.: (96) 310-844/125 • Vercomp Kft. Tel.: (96) 31
• **Hódmezővásárhely:** Delfin Computer Informatikai Rt. Tel.: (62) 346-728 • **Kaposvár:** Multi-Soft Kft. Tel.: (82) 312-512 • **Kecskemét:** Microsystem-Kecskemét Kft. Tel.: (76) 485-054 • T&F Computers Kft.
(76) 493-394 • **Mátészalka:** Wantex Kft. Tel.: (44) 313-555 • **Miskolc:** Bell Computer kft. Tel.: (46) 303-241 • Műszertechnika-Miskolc Kft. Tel.: (46) 411-619 • Profésszionál-Miskolc Kft Tel.: (46) 412-054 • S
Computer-M. Tel.: (46) 344-829 • **Pécs:** SZÜV-Computer-M. Tel.: (72) 252-355 • **Szeged:** 3S Computer Szövetkezet Tel.: (62) 326-247 • Digital kft. Tel.: (62) 490-553 • Montana Kft. Tel.: (62) 432-043 • Summa
Kft. Tel.: (62) 477-583 • **Székesfehérvár:** Flash4 Kft. Tel.: (22) 320-950 • **Szolnok:** Karakter Kft. Tel.: (56) 422-806 • SZÜV-Computer-M. Tel.: (56) 423-330 • **Szombathely:** Flag Kft. Tel.: (94) 322-134 • S
Computer-M. Tel.: (94) 324-251 • **Tapolca:** Balaton Elektronika Kft. Tel.: (87) 412-564 • **Tatabánya:** Alfadat Kft. Tel.: (34) 310-729 • Korall Számítástechnikai és Kereskedelmi Kft. Tel.: (34) 317-181 • **Veszprém:**
Elektronik Kft. Tel.: (88) 428-490 • F&F Kft. Tel.: (88) 324-565 • Műszertechnika-Veszprém Kft. Tel.: (88) 426-506 • **Compaq Disztribútorok**: CHS Hungary Kft. Tel.: 302-4224 • Computer 2000 Tel.: 267-1
Számalk Rendszerház Rt. Tel.: 203-0367 • **Független Compaq szervíz partnerek**: Rikvest Kft. Tel.: 204-2021 • Digital Multivendor Customer Services Tel.: 166-9428



Tavaszi szél ... most apályt hoz

Az ORACLE HUNGARY május 20-ig tartó, tavaszi akciójával támogatja a **szoftverfejlesztő cégek** és a **vállalati szoftverfejlesztők** munkáját!

Web Developer, WebServer, InterOffice


Oracle Developer


Oracle WebServer


Oracle InterOffice

A kereskedelmi csomagba kötött három termék új verziói alkotják az ún. **Web Developer készletet**, amellyel Internet/intranetes, Web-alapú alkalmazások fejleszthetők.

A **Web Developer** egy magas szintű, webes alkalmazásfejlesztő eszköz, mellyel a WebServeren futtatható alkalmazások készíthetők.

A **WebServer** lehetővé teszi az adatbázis alapú webes alkalmazások használatát.

Az **InterOffice** irodaautomatizálási alkalmazás elvégzi az irodákban, munkahelyeken szükséges elektronikus levelezési, dokumentumkezelési, ütemezési, naplózási, stb. feladatokat.

A Web Developer készlet ára: ~~1 829 000 Ft*~~ 1 399 000 Ft*

25 % kedvezmény

Power Objects 2.1, Database Designer


Oracle Power Objects


Oracle Database Designer

A kedvelt **Power Objects** gyorsfejlesztő eszköz legújabb verziója, melynek segítségével webes alkalmazások is készíthetők, most sok kiegészítő eszközzel kerül piacra.

Kiegészítőként az Oracle egy eddig hiányzó, az adatstruktúrákat grafikusan tervező, modellező szoftvert kínál. A **Database Designer** Oracle adatbázist, és ODBC-csatolón keresztül elérhető más adatforrásokat is képes használni (ASCII, Excel, Foxpro, dBase, stb.).

A két szoftver együttes ára: ~~591 850 Ft*~~ 440 000 Ft*

25 % kedvezmény

Információ: **Gruhala Péter**
Ingyenesen hívható telefonszám:
00-800-12000

*Az árak az ÁFÁ-t nem tartalmazzák!

ORACLE®
Az Információs Kor valóra válik!

ORACLE HUNGARY
1123 Budapest, Alkotás u. 17-19.
Telefon: 214-0050, fax: 214-0070
http://www.oracle.com



INFORMÁCIÓKÉRÉS: 0536 ▲